[illegible] 8 [illegible]

105I-H-11

Y3.N88:25/1589

NUREG/CR-1589
SAND80-1626
RS

Pathfinding Simulation (PATHS) User's Guide

COMPLETED

Dennis Engi, Jay S. Shanken, Sandia National Laboratories
P. W. Moore, Tech Reps, Inc.

Printed September 1981

Prepared for
US NUCLEAR REGULATORY COMMISSION

Sandia National Laboratories

[illegible] (2-3-80)

NOTICE

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, or any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for any third party's use, or the results of such use, of any information, apparatus product or process disclosed in this report, or represents that its use by such third party would not infringe privately owned rights.

Available from
GPO Sales Program
Division of Technical Information and Document Control
U.S. Nuclear Regulatory Commission
Washington, D.C. 20555
and
National Technical Information Service
Springfield, Virginia 22161

NUREG/CR-1589
SAND80-1626
Unlimited Release

PATHFINDING SIMULATION (PATHS) USER'S GUIDE

D. Engi
J. S. Shanken
Sandia National Laboratories
Albuquerque, NM 87185

and

P. W. Moore
Tech. Reps., Inc.
Albuquerque, NM 87110

Date Published: September 1981

Sandia National Laboratories
Albuquerque, New Mexico 87185
operated by
Sandia Corporation
for the
U.S. Department of Energy

Prepared for
Division of Facility Operations, Safeguards Branch
Office of Nuclear Regulatory Commission
Washington, D.C. 20555
Under Interagency Agreement DOE 40-550-75
NRC FIN No. A1060

BLANK PAGE

ABSTRACT

The interactive Pathfinding Simulation (PATHS) computer program treats the stochastic shortest route problem through the use of a Monte Carlo technique. This programm is described in its application to the evaluation and design of facility security systems that are used for protection against sabotage and theft. For complex facilities, the number of distinct routes that are available to an intruder intent upon sabotage or theft can be in excess of 10^{100}. By making use of an efficient algorithm for focusing upon only the relatively critical routes (from a security viewpoint), PATHS obviates the need to identify and evaluate each and every complete potential route. As a result, the PATHS procedure permits facility analyses that would otherwise be beyond the capabilities of current generation computers. Input requirements, input modifications, output options, and interpretation of the output are presented, and examples are given of simulations that apply to a hypothetical facility.

BLANK PAGE

CONTENTS

Page

SUMMARY 11

1. INTRODUCTION 13

2. DESCRIPTION OF PATHS 17
  Facility Graph 17
  Fundamental Aspects of PATHS 20
    Measure and Strategy 20
    Path Generation (Phase 1) and Path Evaluation (Phase 2) 22
    Phase 1 and Phase 2 Estimates 23
    Conditional and Unconditional Probabilities 25
    Path Criticality Estimates 26
    A Representation of the PATHS Treatment 28
    Sample Size 31

3. CODE DESCRIPTION 33
  Principal Features--Flow Diagram 33
  Discussion of Subroutines 33
    Subroutine DFLT (Distributions) 36
    Subroutine MOD (Type Codes) 38
    Subroutine PATHS (Storing Information) 38
    Subroutine PDIJK (Detection Probabilities) 39
    Subroutine MNDPT1 (Monte Carlo Simulation) 41
    Subroutine PATHS (Collection of Phase 1 Information) 44
    Subroutines THYME and EZ (Path Evaluation) 46
    XPATH (Computation of Statistics and Output) 48

4. EXECUTION OF PATHS 53
  Input to PATHS 53
    Accessing the PATHS Procedure File 53
    User-Specified Input 54
  Output of PATHS 67
    PATHS Summary 67
    Path Description 73
    Path Ranking/Interruption or Time Information 73
    Node Ranking Information 75
  Examples and Discussion 75
    Interval, Precision, and the Security Protection System 75

CONTENTS (Continued)

Page

Normal and Extended Content in the Path Ranking Information 79

Comparison of Multiple and Single Target Simulations 79

Output Resulting from Changes in MRUN and SEED 83

Interpretation of Extended Path Ranking Information 83

Factors Influencing the Number of Paths Generated 87

Analysis of Variance Resulting from Changes in MRUN and SEED 90

Summary of Examples and Discussions 91

5. OPERATIONAL CONSIDERATIONS 93

Run Times 93

Storage Requirements 99

6. CONCLUSIONS 101

7. REFERENCES 103

8. APPENDIX A -- Expected Values and Variance of the Interruption Probabilities 107

APPENDIX B -- Lower and Upper Bounds to the Minimum Interruption Probability 115

APPENDIX C -- Relationship between Conditional and Unconditional Interruption Probabilities 119

APPENDIX D -- Maximum Variance of a Random Variable with Interval [0,1] 121

APPENDIX E -- Determination of the Minimum Number of Pathfinding Replications 123

APPENDIX F -- Descriptions of Prior Probability Distributions Used in PATHS 129

APPENDIX G -- Cumulative Probability of Detection for a Path 133

APPENDIX H -- Monte Carlo Sampling of Prior Probability Distributions 137

APPENDIX I -- A PATHS Listing 141

9. INDEX 183

10. GLOSSARY 187

LIST OF FIGURES

Figure

1 Facility Layout - Level 3 18

2 Facility Representation - Level 3 19

3 Strategy Corresponding to Combination of Minimum Detection and Minimum Time Measures 21

4 A Representation of the PATHS Treatment 29

LIST OF FIGURES (Continued)

Figure		Page
5	Statistical Tolerance Intervals and Security Threshold Values	30
6	Generalized Flow Diagram for PATHS Showing Functions of Principal Subroutines	34
7	Facility Graph and Interruption Probabilities for Two Runs	45
8	PATHS Input for an Interruption Simulation Using the Extended Mode	55
9	PATHS Input for an Interruption Simulation Using the Brief Mode	58
10	PATHS Input for a Time Simulation Using the Extended Mode	59
11	PATHS Input for a Time Simulation Using the Brief Mode	61
12	PATHS Output for an Interruption Simulation Showing both Standard and Extended Content	68
13	PATHS Output for a Time Simulation Showing both Standard and Extended Content	70
14	PATHS Output for a Single Target Interruption Simulation	76
15	PATHS Output for a Single Target Time Simulation	77
16	PATHS Output for a Two-Target Time Simulation	80
17	PATHS Output for a Two-Target Interruption Simulation	82
18	PATHS Output for a Two-Target Interruption Simulation with Increased Pathfinding Replications	84
19	PATHS Output for a Two-Target Interruption Simulation with Changed Seed	85
20	PATHS Output for a Two-Target Interruption Simulation with Changed Seed and Increased Pathfinding Replications	88
21	ANOVA Tables for Testing the Hypothesis that the Observables are Independent of the Seed and the Number of Pathfinding Replications (≥ 117)	91

LIST OF TABLES

Table		
1	Run Times (Seconds) for Two Facilities and the Product $L = N_p \cdot IRUN$ for the Interruption Measure Using Response Time = 1000 Minutes and Targets 1 (Facility 1) and 278 (Facility 2)	95
2	Comparison of Run Times (Seconds) to $K = N_T \cdot MRUN$ for Interruption and Time Measures Using Facility 1 and Targets 1 and 1 plus 3	96
3	Comparison of Run Times (Seconds) for Two Facilities Using the Interruption Measure and Response Times = 3 Minutes	97

BLANK PAGE

SUMMARY

The interactive Pathfinding Simulation (PATHS) code is a computerized method developed for treating the stochastic shortest path problem. PATHS was developed under contract to the U.S. Nuclear Regulatory Commission as a part of the methodology for the evaluation of the effectiveness of nuclear facility safeguards. An important part of this evaluation is the identification and quantitative appraisal of critical paths within a facility. These are paths that, if traversed by an intruder intent upon the sabotage or theft of a target(s), are most favorable to the intruder in terms of some measure such as the minimum time for traversal or minimum probability of detection.

PATHS uses a Monte Carlo simulation technique to generate these paths and to assess their relative criticality. In addition to descriptions and assessments of these generated paths, the PATHS procedure provides estimates of the lower and upper bounds to the minimum overall security of a facility that are derived from the estimates of path criticality.

This document describes the fundamental aspects of the PATHS procedure and instructs the user in the execution of PATHS on an interactive terminal. Examples of simulations using PATHS are provided and discussed in order to show what inputs are required, how these inputs are entered in the interactive mode, what inputs may be modified and how, and what output can be obtained and how the ouput may be interpreted.

12

BLANK PAGE

1. INTRODUCTION

The Pathfinding Simulation (PATHS) Code provides a Monte Carlo technique for treating the stochastic shortest route problem. This problem is to find the shortest path(s) from any given point to any other given point in a network of points that are connected by path segments the "lengths" of which correspond to the values that can be taken on by one or more random variables. Points may simply be locations or, like path segments, they may have the property of "length". In the latter case, random variables also represent the values for the "lengths" of these points. Point and path segment "lengths" can be defined in terms of appropriate measures, e.g., distance, time, loss, risk, or detection probability. Values for these "lengths" may be fixed or they may be selected by random draws from probability distributions that are appropriate for the characterization of the random variables used in each problem.

If a network of points and path segments serves as a suitable model for a physical system, then PATHS may be used to identify and evaluate the possible paths that are shortest with respect to some measure or combination of measures in this system. A simple application of PATHS would be the determination for a given year of the route of shortest time by air from Poughkeepsie, New York to Eureka, California based on the allowable interconnections between all cities served by air, the distribution of flying times associated with each leg, and the distribution of layover times at each intermediate city. A number of routes could be identified, but on the average (over schedules for a year), one route, the stochastic shortest route, would be preferred over all the others.

Although the PATHS procedure has application to a large class of stochastic shortest path problems, the impetus to its development was the Safeguards Automated Facility Evaluation (SAFE) methodology and it is from this perspective that PATHS will be discussed in this document.

In applying SAFE, the process for the evaluation of physical protection systems at nuclear facilities entails five major phases: (1) facility characterization, (2) facility representation, (3) component performance, (4) adversary path analysis, and (5) effectiveness evaluation. Within this methodology, PATHS provides a procedure for adversary path analysis and path evaluation when the goal of an adversary is the sabotage or theft of a vital component (target) or components within a facility. For targets specified by the user, PATHS generates those paths to the targets that, if selected by an adversary, would most likely lead to adversary success. Paths having this property are called "critical paths," and they are regarded as critical with respect to various measures. Measures that are available within PATHS include minimum time, minimum detection probability, and minimum interruption (by security guards) probability. The appropriateness of these measures is based ultimately on the reasonableness of the assumption that an adversary will, indeed, seek to move along paths that minimize traversal time, along paths that minimize detection probability, or along paths that minimize interruption probability (i.e., that minimize both traversal time and detection probability).

Each execution of the PATHS procedure is regarded as a simulation in which all paths in the facility are examined, but only those paths that are critical with respect to some measure are generated and identified. The PATHS procedure, which uses a Monte Carlo approach for the generation of the critical paths, gives a description of each path that is generated and provides estimates of the criticality of each path. This criticality is expressed as a rank, or relative frequency of generation, for each path and as a probability of interrupting or detecting the adversary (or, if time is the measure of criticality, a time for the adversary to traverse the path). Both the rank and the interruption or detection probability (or traversal time) serve to indicate the extent to which a path is critical.

In addition to the path information, each simulation ranks the components (nodes) located along potentially critical paths by the relative frequency of their occurrence among all generated paths. Moreover, each simulation provides estimates (over all generated paths) of overall facility performance as two statistics: a lower bound, which is an average of minimum interruption or detection probabilities (or average of minimum traversal times), and the upper bound, which is the minimum of an average of interruption or detection probabilities (or minimum of an average of traversal times).

Essentially, the adversary path analysis and evaluation procedure that is carried out by PATHS performs three important functions:

- First, potentially critical paths are generated and identified.
- Second, generated paths are evaluated, paths and path components are ranked and estimates of bounds for facility performance are computed.
- Third, randomness in the time parameters is accounted for through the use of five prior probability distributions in a stochastic approach that incorporates both Monte Carlo and analytic probabilistic techniques.

In terms of these functions, PATHS is superior to previously developed path analysis and evaluation procedures that have been used in the evaluation of sophisticated nuclear facility protection systems. With respect to the first function PATHS is similar to the Minimum Detection Probability and Minimum Time (MINDPT)[2] procedure and is unlike the Estimate of Adversary Sequence Interruption (EASI)[3,4] procedure, which requires prior specification and description of potentially critical paths. All three procedures, PATHS, MINDPT, and EASI are similar with respect to the second function but, with respect to the third, the three procedures are notably different. The MINDPT procedure is completely deterministic and therefore dependent upon prior knowledge of a single representative value for each of the input parameters. The EASI procedure treats detection probability parameters as fixed at their mean or expected values and treats time parameters as random. The random parameters are handled in a strictly analytic probabilistic manner that assumes a prior probability distribution and requires the specification of the means and standard deviations of these parameters. Although the treatment of detection probability parameters is equivalent to that in EASI, the PATHS procedure obtains random values for the time parameters using the Monte Carlo technique to sample user-specified prior probability distributions that need not be the same for each random variable.

The PATHS procedure, then, constitutes a further development and generalization of the best attributes of MINDPT and EASI. Because of its enhanced capacity for both path analysis and evaluation, PATHS can function not only as a modular component of a more extensive facility evaluation procedure such as SAFE but also as an independent analysis and evaluation procedure.

The purpose of this document is to provide a guide for the use of the PATHS computer program on an interactive terminal and to aid in the interpretation of the resulting output. The concepts and methodologies that are essential to the PATHS treatment are discussed in order to assist the user in understanding the basic features of this program. Descriptions of the input parameters, the values of which are specified by the user or provided to the program from existing files, are given. Instruction for user interaction with the program is provided by examples of simulations that use various inputs. The meaning and the significance of the outputs that result from these example applications are discussed with regard to the statistical treatment that is used in PATHS and with regard to changes that can be made in the inputs.

2. DESCRIPTION OF PATHS

Facility Graph

In order to describe the PATHS procedure, it is useful to understand how the physical facility appears in the PATHS model. A suitable facility description is achieved in a three-step process. First, the facility is characterized in terms of the facility layout, including barriers, access points, floor levels, and interconnections. Targets and vital areas are identified using codes such as the Set Equation Transformation System (SETS),[5] and locations and descriptions of physical protection system components are obtained from facility documentation or site visits. The characteristics of the guard (response) force may be similarly determined. Second, a computer representation of the facility is produced using this information. Figures 1 and 2 show the reduction of a facility layout or blueprint to a simple representation of the facility. For a sabotage scenario,* the complete facility representation in PATHS is an undirected sabotage graph[6,7] composed of labeled (numbered) nodes and arcs that are weighted by detection probabilities and traversal times. Step three provides the node and arc weights in the form of detection probabilities and traversal times through selection of the performance of the components within the physical protection system. Component performance may be determined from data obtained from the Safeguards Engineering and Analysis Data-Base (SEAD).[8]

The sabotage graph can be constructed to describe a facility in whatever detail the user requires. Nodes, representing boundaries (perimeter gates or fences), penetrable barriers (sensors and alarms, doors, windows, walls, vaults) and targets (critical material or vital equipment), may be placed at all important locations. Arcs are placed on the graph to represent the physical paths from one location to another.

---

[*] This User's Guide can assume a sabotage rather than a theft scenario without compromising the generality of the discussion.

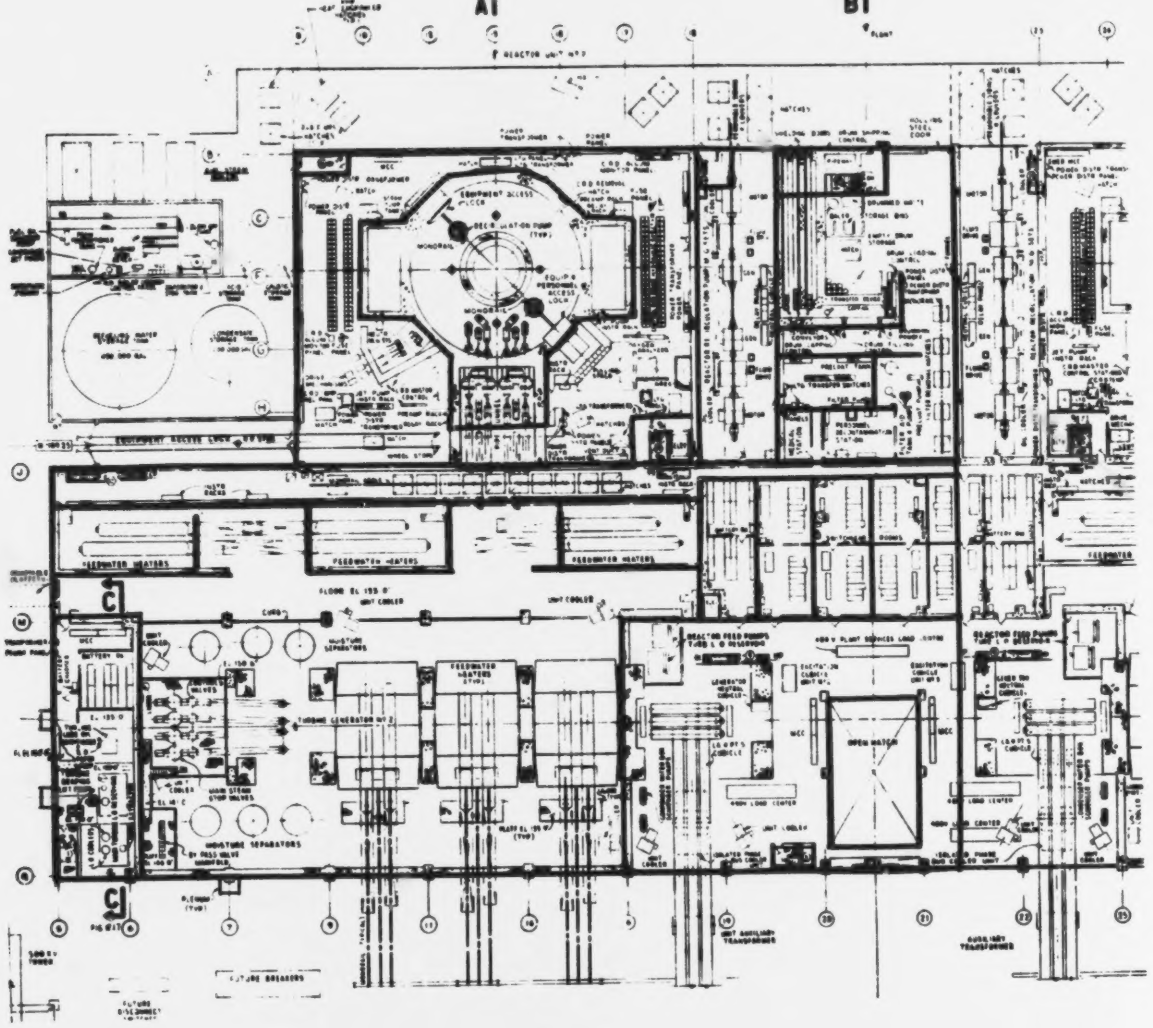


Figure 1. Facility Layout - Level 3

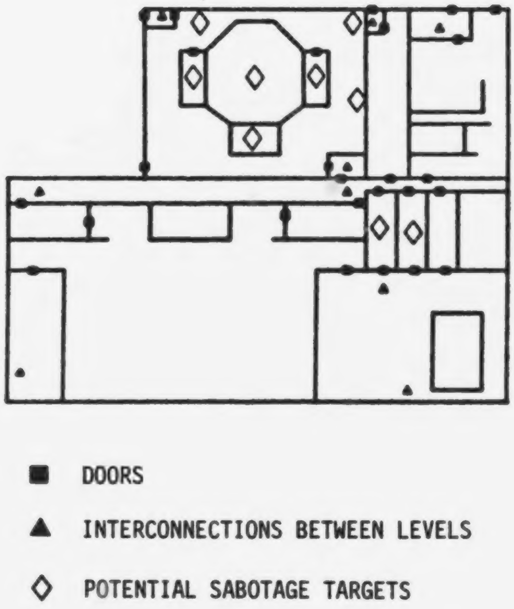

■ DOORS
▲ INTERCONNECTIONS BETWEEN LEVELS
◇ POTENTIAL SABOTAGE TARGETS

Figure 2. Facility Representation - Level 3

For PATHS, each node and arc is labeled and also weighted with the input values for a detection probability and a traversal time. Given the facility, these weights, except for the arc detection probabilities which are computed from arc traversal times, are available to the user as default values in an input file (Tape 20) that is processed at the beginning of the PATHS procedure. For detection probability parameters, these initial values are regarded as the means or expected values corresponding to a deterministic distribution of these variables; for time parameters, the initial values are treated as the modes, or most probable values, corresponding to any of five different user-specified probability distributions. During the execution of PATHS, the prior distributions and the values for the time parameters can be modified by the user. The expected values for the detection probability parameters can also be altered, but these parameters remain deterministic. The significance of this constraint will be made clear later (see p. 22 under Fundamental Aspects of PATHS and Appendix A).

The source material upon which PATHS acts, then, consists of the labeled and weighted facility graph along with certain additional required user inputs. The nature of these additional inputs will be discussed in a following section.

Fundamental Aspects of PATHS

Given the requisite source material, the PATHS procedure generates and evaluates only those paths that are critical to the security of the facility. It has been estimated that some facilities have in excess of 10^{100} distinct paths. Were every possible path generated and evaluated with current generation computers, the task would be impossible. Therefore, PATHS uses an efficient algorithm to find only relatively critical paths, and the resulting output represents an analysis of the facility that estimates worst case (from a security viewpoint) bounds to adversary strategies and actions.

Measure and Strategy -- The PATHS pathfinding algorithm, essentially a Dijkstra[9] search technique modified by Yen[10,11], permits examination of alternative adversary strategies. The strategies correspond to the minimization of either one or both of two measures in the PATHS procedure. The first measure, traversal time, represents a strategy by which the adversary, using force, attempts to penetrate to the target as rapidly as possible. The second measure, detection probability, represents a strategy by which the adversary, using stealth and deceit, tries to reach the target with a minimum probability of being detected. A more general, mixed strategy, illustrated in Figure 3, is represented by a combination of the two measures. In this case, the adversary tries to minimize the probability of interruption by the security force by minimizing detection probability up to a given locus from the target, and thereafter, by minimizing time for traversal from a point on the locus to the target. This locus is a set of points positioned at a guard response time away from the target; that is, these points can be reached by alerted security personnel in, for instance, RT minutes. Outside the response time locus, the adversary can always be interrupted if detected; inside the locus, the detected adversary can avoid interruption if his traversal time is less than RT.

To treat the mixed strategy then, PATHS requires the explicit specification of interruption probability as the measure, and two modified Dijkstra searches are carried out.* For each generated path, the effect is as if one segment were examined with respect to minimum detection

---

*These searches are generalizations of the deterministic pathfinding algorithm that is used in MINDPT[2].

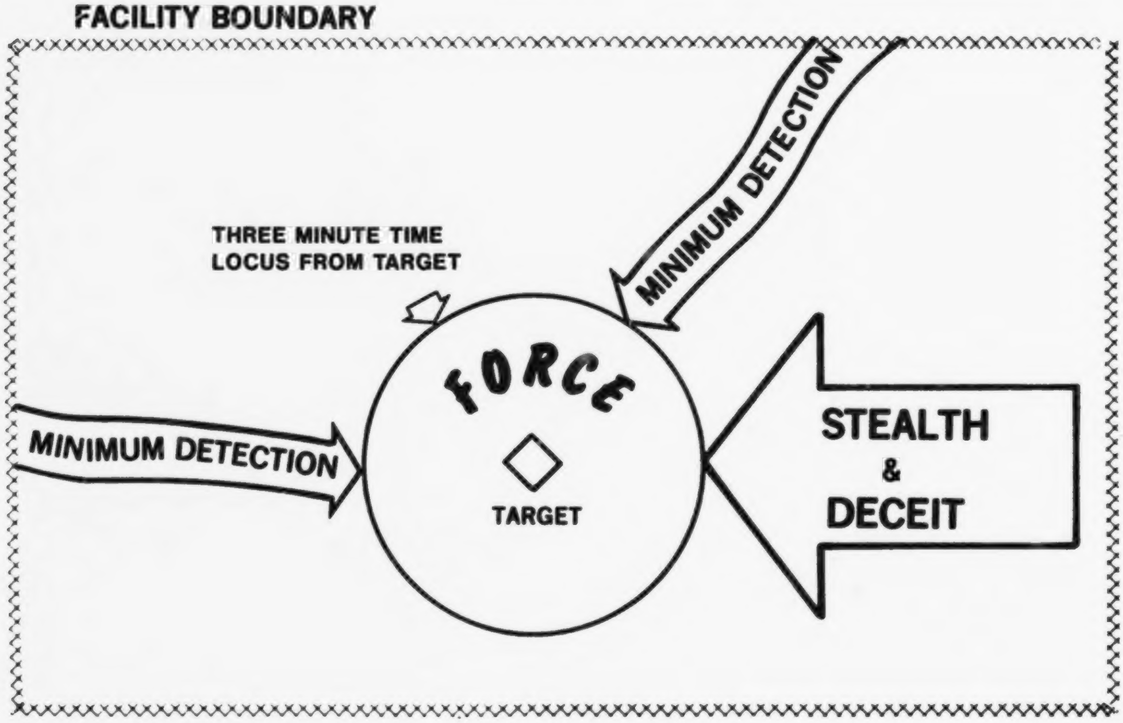


Figure 3. Strategy Corresponding to Combination of Minimum Detection and Minimum Time Measures

probabilities specified for each node and arc, finds minimum detection paths from off-site to every node in the graph. The second search determines the locus of points at a corresponding guard response time away from each specified target and, using node and arc traversal times, finds minimum time paths between each target and its corresponding set of locus points. The results of these searches are combined to yield critical paths that minimize both traversal time and detection probability.

It should be noted that PATHS does not require the explicit specification of detection probability as a measure. Indeed, the pure strategies corresponding to either minimum detection probability or minimum traversal time can be regarded as special cases of the general strategy for which interruption probability is the measure. By setting the guard response time to zero, paths are generated that minimize only detection probability; by setting the response time to a very large value, paths are generated that minimize only traversal time. Nonetheless, when the measure is minimum traversal time, it must be specified explicitly.

Path Generation (Phase 1) and Path Evaluation (Phase 2) -- PATHS separates the tasks of path analysis and path evaluation into two distinct phases: a path generation phase (Phase 1) and a path evaluation phase (Phase 2). Phase 1 searches for and finds paths that are minimal and therefore critical with respect to the specified measure. Phase 2 takes those paths that have been generated in Phase 1 and evaluates them, i.e., makes a quantitative assessment of each path with respect to the measure.

In order for PATHS to carry out these tasks, the user must supply values for the input parameters that are representative of the facility. For any real facility, uncertainty will exist with regard to the true values of these parameters. PATHS allows the user to account for this uncertainty by assigning prior probability distributions to the input parameters. This capability allows the user to treat most inputs, specifically the time parameters, as random variables rather than as fixed quantities.

It can be demonstrated (see Appendix A) that in the case of the detection probability parameters, only the means or expected values of the prior distributions need be considered in the PATHS model. Therefore, without loss of generality, the input values of the detection probability parameters are regarded as if they were the means of prior distributions, but they are also fixed at these values throughout a given simulation. These values may, of course, be the default values or user-specified (modified default) values. Whichever is the case, they are unchanged during both the path generating phase and the path evaluation phase, and this results in the saving of time for computation and in greater precision for all estimates (statistics) obtained using PATHS. (See Appendix A)

Regardless of the measure that is specified, the random variables are handled in the path generation phase by Monte Carlo techniques that consist of repeated sampling of the prior distributions. During a single simulation, this sampling procedure is carried out many times, i.e., for many runs or replications. Each replication consists of a new realization (set of values) for the node and arc traversal times and guard response times. A critical path is generated for each realization, and the resulting path may or may not be identical to a path that was generated for a previous realization. Provided that the number of replications is sufficiently large, the probability of overlooking a potentially critical path is small. Indeed, if the probability of generating a path on a single replication or run is as small as 0.05, then the probability that it is never generated in MRUN = 100 replications is less than 0.01. This follows from the application of the probability function for binomial trials. As a consequence, the Monte Carlo approach not only serves to account for the inherent randomness in the values of the time parameters but also serves to create reasonable confidence that no critical path has been overlooked.

Monte Carlo techniques are also used in the path evaluation phase for handling random variables when interruption (or detection) probability is the measure. In this case, Phase 2 consists of IRUN replications or runs resulting in IRUN realizations of the node and arc traversal times and guard response times for each path that was generated in Phase 1. When time is the measure, Phase 2 uses an analytic probabilistic technique similar to that used in EASI. Having generated minimum time paths using Monte Carlo techniques in Phase 1, it is sufficient to evaluate these paths in terms of the means and variances of the prior probability distributions that have been assigned to the input parameters. Whatever the Phase 2 technique, both the path generation phase and the path evaluation phase yield different but complementary estimates of path criticality and facility performance.

Phase 1 and Phase 2 Estimates -- The unknown parameter of the PATHS model is the minimum interruption probability (or minimum traversal time) over all paths, min P. The facility estimates (estimates over all paths) that are obtained from the execution of PATHS provide bounds for min P. As a result, facility estimates can be used to draw conclusions about facility performance with respect to the specified measures by comparing them with some acceptable threshold value, P_T, for the interruption probability (but not the traversal time).

The MRUN Phase 1 runs generate a facility estimate or statistic, LB, that is the lower bound to min P. The lower bound is an estimate for the mean of the least probabilities of interruption (or mean of the least traversal times) along all paths to the specified targets. Hence the lower bound is not always representative of a real path but of a composite path (and target). The IRUN Phase 2 runs generate a facility estimate or statistic, MIN, that is the upper bound to min P. The upper bound is an estimate for the least mean probability of interruption (or least mean traversal time) that obtains for a particular real path (and target) within a set of minimum paths generated in Phase 1.

When time is the measure, it is computationally feasible to generate both a lower and upper bound. When viewed as bounds, however, these estimates are not necessarily useful for drawing conclusions about facility performance. Unlike a threshold value for the interruption probability, a corresponding threshold value for the traversal time cannot be meaningfully specified. If such a value were specified, it would be meaningful only in relation to a guard response time, and the appropriate measure would be interruption probability, not traversal time. Because of this, there is no threshold with which to compare the bounds. What is needed in this instance is simply an estimate of the unknown minimum traversal time with the assurance that the estimate errs on the high rather than the low side. The upper bound or least mean traversal time is such an estimate and this is the Phase 2 facility estimate that is generated when the measure is time. Generating the lower bound would, under these circumstances, be useful only as a check on the Phase 1 sample size (see following paragraphs).

Because min P lies somewhere between the upper and lower bound (see Appendix B); that is $LB \leq \min P \leq MIN$, the facility is not secure if, relative to the desired facility threshold, the upper bound that results is less than the threshold ($MIN < P_T$).[*] The facility is considered to be secure if the resulting lower bound is greater than the threshold ($LB > P_T$).[**] When both estimates bracket the threshold value ($LB < P_T < MIN$), the situation is uncertain. An examination of the variance for the estimate of the lower bound might be useful in this case. If this variance is large, it is possible that a path was generated in Phase 1 for which a realization of the input parameters was statistically unlikely.

In actual applications the indeterminate situation ($LB < P_T < MIN$) does not often arise, but it does occur more often when multiple targets, rather than a single target, are specified. What can arise in actual applications, though, is that the value generated for the lower bound exceeds that for the upper bound. What has occurred is that the Phase 1 sample size, which is normally less than the Phase 2 sample size

---

[*] Specification of the facility threshold is the result of a policy decision on the part of the user; in the context of PATHS, the value of P_T is somewhat arbitrary.

[**] The phrase "considered to be" is used here because there is no assurance that the adversary is neutralized even though the facility is secure with respect to the measure.

(MRUN < IRUN), is simply too small. This can be corrected by increasing the number of Phase 1 runs so that the variance in the estimate of the lower bound is reduced.*

Facility estimates provide information about overall facility performance, but path criticality estimates provide information about each path that is generated and evaluated in the PATHS procedure. Path criticality estimates are obtained for paths generated in Phase 1 by ranking each path in terms of its relative frequency of generation, i.e., the ratio of the number of times a path is generated to the total number (MRUN) of Phase 1 replications. Phase 2 path criticality estimates are obtained for paths generated in Phase 1 by an evaluation that computes an estimate (for each path) of the mean probability of interruption (or mean traversal time) over the total number (IRUN) of Phase 2 replications that are carried out for each path. The rank and estimate of the mean interruption probability (or mean traversal time) that are associated with each path are therefore estimates of path criticality that apply to single real paths. Path components, the nodes that constitute elements of these generated paths, are also ranked and each node rank is the ratio of the number of times the node is generated to the total number (MRUN) of Phase 1 runs or replications.

Conditional and Unconditional Probabilities -- The starting point for any estimate of facility performance or path criticality is the probability of interruption $P_I = P(D,T)$. The independent random variables D and T may take on all values d_1, d_2, d_3, ..., of the detection probabilities and all values t_1, t_2, t_3,..., of the node and arc traversal and guard response times that are consistent with the specified prior distributions. In PATHS, however, the prior distribution for the detection probabilities is deterministic and the values d_1, d_2, d_3,..., are fixed throughout a PATHS simulation (cf. Appendix A). As a consequence, $P_I = P[E(D),T]$, and in both phases only the values of the time parameters are allowed to vary from run to run. With interruption probability as the measure, these values are determined during each Phase 1 and Phase 2 run by sampling the prior distributions to obtain values for the delay and response times.

---

*Unfortunately, the magnitude of the variance (or standard deviation) cannot be used to determine exactly how much MRUN should be increased; the user must simply increase MRUN until LB $\leq$ MIN.

During each run, Phase 1 computes P_c = min P[E(D),T] over all paths in the facility graph. This quantity, P_c, is the conditional probability of interruption, so named because P_c is evaluated and recorded only on the condition that P[E(D),T] is a minimum. A path for which this minimum value applies is said to be "generated," and it is potentially a most critical path. Such a path, out of all possible paths, is found for each run (i.e., for each realization of the time parameters). It need not be distinct from a previously generated path. Indeed, each time a particular path is generated, this fact is noted; the count (the number of times this occurs) divided by the total number of Phase 1 runs is the rank of the path. The rank is an estimate over all generated paths of the probability that the path is most critical (is the minimum path) with respect to the measure specified by the user. In addition, an estimate of the expected value, $E_T(P_c)$, for each path (over those replications that generated the path) is computed. This estimate is the interruption probability (or traversal time) for the path given that the path is a minimum with respect to the measure. The Phase 1 facility estimate, LB, is the arithmetic mean of the conditional probabilities of interruption that have been computed for each and every run; that is, LB, the lower bound, is an estimate of $E_T(P_c)$ over all replications.

Phase 2 uses the Monte Carlo sampling techniques to evaluate the paths generated in Phase 1. For each path, Phase 2 computes P_u = P[E(D),T] without regard to any minimizing condition on every replication or run. The quantities, P_u, are therefore the unconditional probabilities of interruption or simply the probabilities of interruption for the path. For a given path, the expected value, $E_T(P_u)$, over all Phase 2 runs is the mean unconditional probability of interruption for that path and an estimate of $E_T(P_u)$ is obtained as the arithmetic mean of the values, P_u, computed on each and every Phase 2 run. The estimate associated with each path is, in addition to the Phase 1 rank, also an indication of path criticality. In this case, the lower the value of the path estimate, the more critical is the path. The Phase 2 facility estimate, the upper bound MIN, is the estimate of $E_T(P_u)$ that has the lowest value. MIN is therefore equivalent to the Phase 2 path criticality estimate for the most critical path.

Path Criticality Estimates -- The relationship between the Phase 1 and Phase 2 facility estimates, LB and MIN, has been discussed in a preceding section and Appendix B. Appendix C explores the relationship between conditional probabilities (or traversal times) and the Phase 1 and

Phase 2 path criticality estimates. The Phase 2 unconditional interruption probability (or traversal time) for a path is the sum of two terms. The first term is the product of the rank (or probability that the path is most critical) and the Phase 1 conditional probability (or traversal time), i.e, the interruption probability (or traversal time) given that the path is most critical. The second term is the product of the probability that the path is not most critical and the interruption probability (or traversal time) given that the path is not most critical. An equivalent statement in terms of the path criticality estimates, i.e., estimates of rank (r) and $E_T(P_u)$, and the estimate of $E_T(P_c)$ is

$$E_T(P_u) = (r)E_T(P_c) + (1-r)E_T(P_{NOT\ MIN})$$

where the estimate of $E_T(P_u)$ is over all Phase 2 replications for the path and the estimate of $E_T(P_c)$ is over all Phase 1 replications that generated the path. The estimate of $E_T(P_{NOT\ MIN})$ that is found from the expression above is an estimate of the interruption probability (or traversal time) for the path given that it is not the minimum path. This quantity may be of use in assessing the relative criticality of paths that result from a simulation in which, for some or all paths, there is little linear correlation or else inverse linear correlation between the Phase 1 rank and the Phase 2 path criticality estimate for each path.

The extent of the correlation between estimates for all generated paths may be noted from the value of the linear correlation coefficient that is displayed in the output of the "Path Ranking/Interruption" or "Path Ranking/Time information." An unusually small positive value or a negative value may, for example, occur when on the one hand the Phase 1 and 2 criticality estimates for one path are both large, i.e., high rank or probability of being most critical and high probability of being interrupted (or large traversal time), and on the other hand these estimates for another path are somewhat lower, i.e., significant probability of being most critical and relatively low probability of being interrupted (or small traversal time). It may then be difficult to make an assessment of the relative criticality of these paths. In this event, a comparison of the estimates for the interruption probabilities (or traversal times), given that neither path is the minimum path, may indicate which of the two paths, the one that is more likely or the one that is less secure, is more in need of enhanced protective measures.

A Representation of the PATHS Treatment - To put some of the concepts discussed in the last two sections into better focus, consider the situation shown in Figure 4 for a small facility in which only four physical paths to a target are possible. The numbers in the boxes of the left-hand array are the computed interruption probabilities for each path on each run. Although the Phase 1 and 2 runs are statistically identical only for large sample sizes, it will be assumed that this simple array represents the values computed in both phases. To demonstrate the treatment, the number of runs in each phase will be small and equal (MRUN = IRUN = 4). The numbers in the right-hand array represent the statistics that are collected during this simulation.

In Phase 1, only paths 1 and 3 are generated: path 1, three times with rank = 3/MRUN = 0.75, and path 3, once with rank = 1/MRUN = 0.25. This is confirmed by an examination of the columns corresponding to each run. For example, in run 1, P_c = min P_I = 0.7; in run 3, P_c = min P_I = 0.5. Owing to its rank, path 1 is potentially the most critical. $E_T(P_c)$ is estimated by the arithmetic mean of the values of P_c over all four runs. This estimate is the facility lower bound, LB = 0.65. The standard deviation of the (Phase 1) sample, for which LB is the mean, is $\sigma_1 = 0.10$, and $\sigma_{LB} = \sigma_1/(MRUN)^{1/2} = 0.05$ is the standard deviation of the sample mean.

In Phase 2, paths 1 and 3 are evaluated by estimating $E_T(P_u)$ over all runs for each path. The sum across each row divided by IRUN gives the sample mean for path 1, $\mu_u = .78$, and the sample mean for path 3, $\mu_u = 0.8$. The lower value for the path 1 sample mean confirms the Phase 1 path criticality estimate for path 1. (Indeed, the linear correlation coefficient for the Phase 1 and Phase 2 path criticality estimates would, if computed here, be +1. This need not always be the case; in actual practice, the PATHS linear correlation coefficient may vary from -1 to +1.) The standard deviations for the path 1 and path 3 samples are $\sigma_u = 0.15$ and $\sigma_u = 0.20$, respectively. The facility estimate of min $E_T(P_u)$ is, of course, identical to the estimate of $E_T(P_u)$ for Path 1. Hence MIN = 0.78 and $\sigma_2 = 0.15$. The standard deviation of the sample mean is $\sigma_{MIN} = \sigma_2 /(IRUN)^{1/2} = 0.08$.

For this simple facility, path 1 is unambiguously the most critical path, and the unknown facility minimum, min P, lies somewhere between and including LB = 0.65 and MIN = 0.78. Note, however, that path 2 was not generated; had it been generated, the estimate of the upper bound

Interruption Probabilities, P_I

	Phases 1 and 2			
RUN / PATH	1	2	3	4
1	0.70	0.70	1.00	0.70
2	0.80	0.80	0.60	0.80
3	0.90	0.90	0.50	0.90
4	0.90	0.90	0.90	0.90
P_C	0.70	0.70	0.50	0.70

Statistics

Phase 1				Phase 2				
RANK	LB	σ_1	σ_{LB}	μ_u	σ_u	MIN	σ_2	σ_{MIN}
0.75	—	—	—	0.78	0.15	0.78	0.15	0.08
{1.00}	—	—	—	(0.75)	(0.10)	(0.75)	(0.10)	(.05)
0.25	—	—	—	0.80	0.20	—	—	—
—	—	—	—	—	—	—	—	—
—	0.65 {0.80}	0.10 {0}	0.05 {0}	—	—	—	—	—

μ_u = arithmetic mean of the (unconditional) probabilities of interruption for path i = 1,4 (if generated).

σ_u = standard deviation of the Phase 2 sample for path i = 1,4 (if generated).

MIN = facility upper bound = min μ_u.

σ_2 = standard deviation of the Phase 2 sample for the minimum path.

σ_{MIN} = standard deviation of the upper bound.

LB = facility lower bound = arithmetic mean of the conditional probabilities of interruption for all paths generated in Phase 1.

σ_1 = standard deviation of the Phase 1 sample.

σ_{LB} = standard deviation of the lower bound.

Figure 4. A Representation of the PATHS Treatment

min P would have been lowered to MIN = 0.75 (Phase 2 statistics in parentheses),and path 2, rather than path 1, might have been regarded as most critical.

The inferences that may be made with regard to facility security depend upon the threshold value, P_T. Figure 5 illustrates the situation for three values of P_T. Given the interval, [0.65, 0.78], and setting aside considerations regarding precision, the facility may be secure when $P_{T1} = 0.60$, but is not secure when $P_{T2} = 0.80$. For $P_{T3} = 0.76$, the situation is, in fact, indeterminate. If the accepted hypothesis is that the facility is secure because $P_{T3} < \text{MIN}$, a Type II error has occurred. This hypothesis would have been rejected had path 2 been generated, for then $P_{T3} > \text{MIN}$. If precision in the estimates is considered, then the standard error for LB = 0.65 is 0.05 and for MIN = 0.78 it is 0.08. The situation then for all three threshold values is at best marginal and at worst indeterminate.

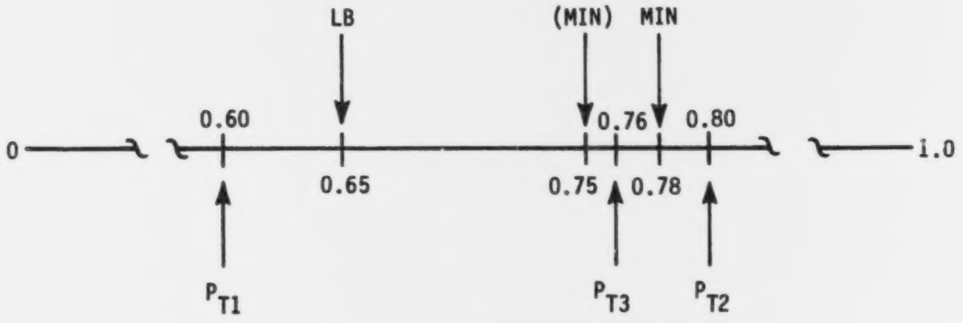


Figure 5. Statistical Tolerance Intervals and Security Threshold Values

To minimize the interval of indeterminacy and improve the precision of the estimates, the number of runs in both phases should be reasonably large. In a PATHS simulation for which MRUN = 500 and IRUN = 1000, the interval that resulted was [0.75, 0.76] with standard errors of 0.01 and 0.01. Hence, both the interval and the probability of overlooking a path (such as path 2) can be relatively small. The interval, of course, is also a random variable that is a function of LB and (MIN - LB) i.e., position and size. Because the facility minimum, min P, is bounded by [LB, MIN], the expected value and variance of min P should be a good indication of interval position and size. If the expected value of min P is indeed P, then the variance of P is at most P(1-P)(See Appendix D) and the variance is small where P is close to 0 or 1. As a result, the interval (and the values indicating the precision of the estimates, LB and MIN, as well) may be expected to be relatively small at the extremes. In particular, if it is positioned in the region of P_T, the interval should be small where the more interesting threshold values, say $P_T > 0.9$, are explored.

This suggests a method of approach for the examination of a facility. If the interval, as determined by an initial pair of facility estimates, falls in a region well below comfortable threshold values, additional simulations incorporating an upgrading of the components of the security system (increased probability of detection, shorter guard response times, etc.) will increase the precision of the estimates, reduce the size of the interval, and move the interval position closer to the threshold value. Because the interval is smaller, the potential indeterminacy is reduced and more definite inferences can be made.

Sample Size -- In the previous subsection, reference was made to a simulation in which the sample sizes were MRUN = 500 and IRUN = 1000. An identical simulation with but one change, MRUN = 117, was carried out. The result of this simulation was LB > MIN. That this is a possibility and why it can occur in actual application has already been mentioned. An example from Figure 4 though can be used to illustrate this occurrence. Consider path 2 of Figure 4 and assume that it is the only path generated. Assume also that IRUN = 4 but that now MRUN = 2 corresponding to just runs 1 and 2. The Phase 1 statistics (bracketed) give rank = 1 and LB = 0.80 (since P_c for each run is 0.80). The Phase 2 statistics (in parentheses) give MIN = 0.75 and consequently LB > MIN.* The estimates of precision do not help to indicate how much MRUN should be increased, so MRUN is increased using repeated trials until LB $\leq$ MIN and the interval is reasonably constant.

Two considerations dictate the choice of sample size. First, the sample sizes for both phases must be adequate to represent the population of interruption probabilities or traversal times and to provide reasonable precision in the estimates obtained. Second, the Phase 1 sample size must be large enough to make small the probability of overlooking a potentially critical path. Normally, attendance to the first consideration is sufficient to ensure that $LB < MIN$ and that the probability of overlooking a path is small. For Phase 2, $IRUN \geq 1000$ is satisfactory, and a comparable MRUN would also be satisfactory. It is usual, however, to specify MRUN initially by using the second consideration alone. Under these circumstances, it is often possible to meet the demands of the first consideration, to have $LB < MIN$, and to save time required for computation. For the Phase 1 sample then, the suggested number of runs is $MRUN = 117$. If this sample size is specified, it is possible to state: given that each path in a set of paths has a probability $\geq p$ of being the minimum interruption (or minimum traversal time) path, then if $p = 0.05$, there is a probability of at least 0.95 that the members of the set will be generated one or more times (see Appendix E). Indeed, if $MRUN = 117$ and $p > 0.05$ then the set is more exclusive, but the members have even greater probability of being generated.

---

*When only one path is generated, the ratio of the lower to upper bound is expected to be at most 1, providing the Phase 1 and 2 samples are statistically equivalent. Of course, if they are not equivalent (as this example illustrates), it is possible to observe LB > MIN.

In actual application, of course, PATHS is not limited to the consideration of $\leq 1/p$ paths but considers all N physical paths in the facility from which a critical set is generated. The selection of a value for p is a device for the demonstration of the path generating capability of PATHS. $MRUN = 117$ must be regarded as a time-saving specification that must be traded off to obtain better facility estimates and better precision when the need arises.

3. CODE DESCRIPTION

Principal Features--Flow Diagram

The main features of the PATHS code are shown in the generalized flow diagram of Figure 6. The names of the principal subroutines, as they appear in the complete code listing,* are capitalized and underscored for each block that describes the subroutine's function. It is apparent that the main routine is XPATH, which controls input and output, data conditioning and modification, and the transfer to and from path generation (Phase 1) operations (PATHS, PDIJK, and MNDPT1) and path evaluation (Phase 2) operations (EZ and THYME).

Discussion of Subroutines

Initially XPATH (1) accepts the facility graph inputs and required user inputs, (2) lists (displays) certain data which echo the inputs, and (3) clears working storage and sets initial conditions that are appropriate to the inputs. (The exact sequence in which these functions are carried out can be discerned from the complete listing.)*

The facility graph inputs are read from a file, Tape 20, that contains node and arc traversal times, node detection probabilities, and numbers used for internal bookkeeping that relate to the organization and size of the graph. User inputs are requested by XPATH; some specific examples include:

1. Mode of input--brief or step-by-step guidance (extended mode),
2. Measure to be used--time (T) or interruption probability (I),
3. Node detection type--detection probability is step (S) or linear (L) function of node penetration,
4. Targets to be examined--how many, labels for, and response times for,

---

*See Appendix I.

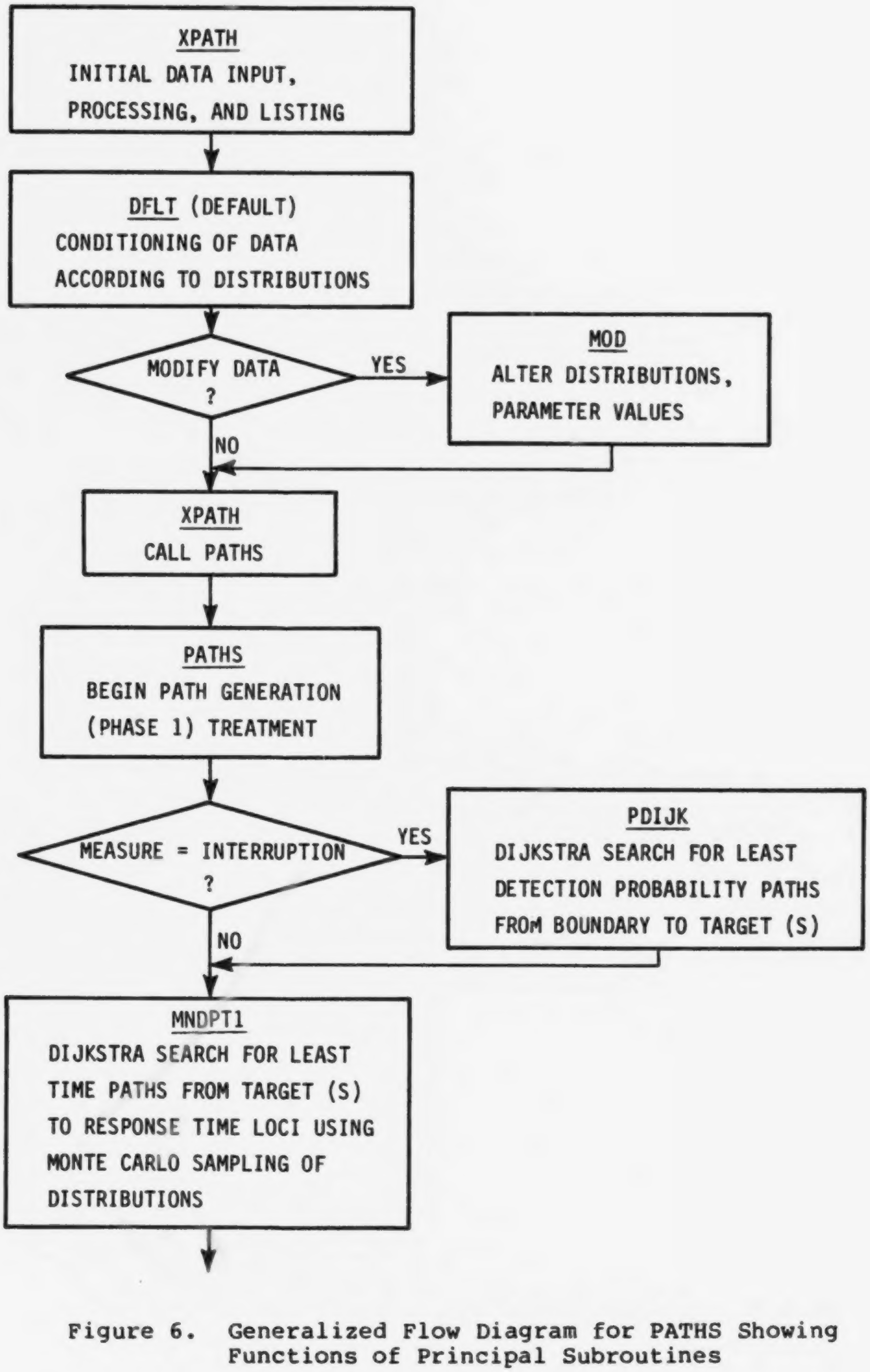

XPATH
INITIAL DATA INPUT, PROCESSING, AND LISTING
DFLT (DEFAULT)
CONDITIONING OF DATA ACCORDING TO DISTRIBUTIONS
MODIFY DATA ?
YES
MOD
ALTER DISTRIBUTIONS, PARAMETER VALUES
NO
XPATH
CALL PATHS
PATHS
BEGIN PATH GENERATION (PHASE 1) TREATMENT
MEASURE = INTERRUPTION ?
YES
PDIJK
DIJKSTRA SEARCH FOR LEAST DETECTION PROBABILITY PATHS FROM BOUNDARY TO TARGET (S)
NO
MNDPT1
DIJKSTRA SEARCH FOR LEAST TIME PATHS FROM TARGET (S) TO RESPONSE TIME LOCI USING MONTE CARLO SAMPLING OF DISTRIBUTIONS

Figure 6. Generalized Flow Diagram for PATHS Showing Functions of Principal Subroutines

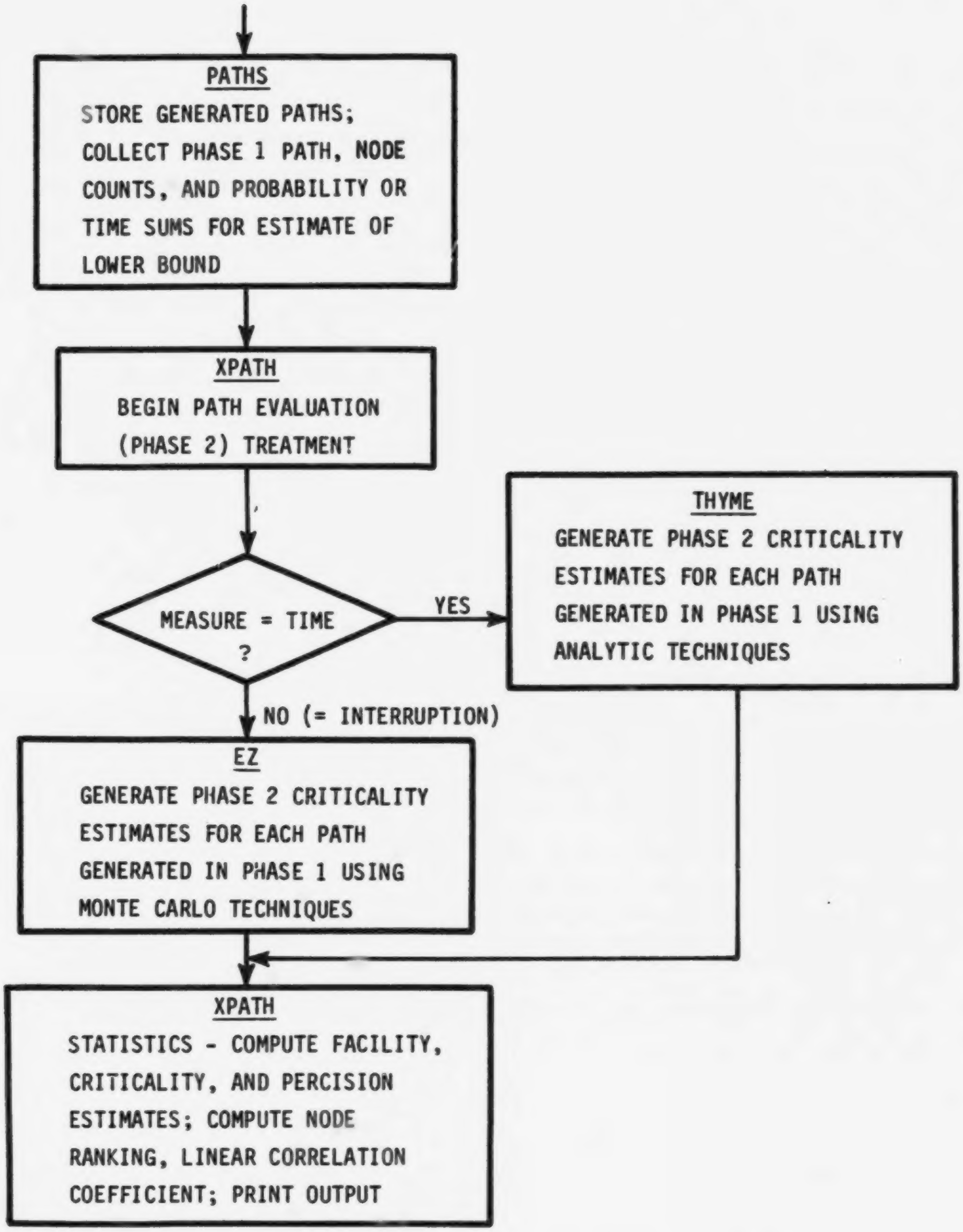

PATHS
STORE GENERATED PATHS; COLLECT PHASE 1 PATH, NODE COUNTS, AND PROBABILITY OR TIME SUMS FOR ESTIMATE OF LOWER BOUND
XPATH
BEGIN PATH EVALUATION (PHASE 2) TREATMENT
MEASURE = TIME ?
YES
THYME
GENERATE PHASE 2 CRITICALITY ESTIMATES FOR EACH PATH GENERATED IN PHASE 1 USING ANALYTIC TECHNIQUES
NO (= INTERRUPTION)
EZ
GENERATE PHASE 2 CRITICALITY ESTIMATES FOR EACH PATH GENERATED IN PHASE 1 USING MONTE CARLO TECHNIQUES
XPATH
STATISTICS - COMPUTE FACILITY, CRITICALITY, AND PERCISION ESTIMATES; COMPUTE NODE RANKING, LINEAR CORRELATION COEFFICIENT; PRINT OUTPUT

Figure 6. Generalized Flow Diagram for PATHS Showing Functions of Principal Subroutines (Concluded)

5. Random number seed,
6. MRUN--number of Phase 1 runs,
7. IRUN--number of Phase 2 runs,
8. Distribution codes--numbers indicating the distributions to be used for node and arc delay times and guard response times, and
9. Type codes--numbers that identify types of input parameters, e.g., node and arc detection probability, node and arc traversal time, guard response time.

XPATH echoes the user inputs and displays certain facility inputs in a listing of (1) the number of target, barrier, and boundary nodes, (2) the number of arcs, and (3) the labels of all target nodes. The listing of these labels aids the user in the accurate specification of targets to be examined.

If the user specifies time as the measure, guard response times for all targets are set equal to BIG (a very large number), and the response time distribution for each target is made deterministic; the node detection type does not enter into consideration. With interruption the measure, the user must specify the node detection type (NTYP). When "S" is specified as the node detection type, the node detection probability is a step function for all nodes. This means that when the guard response time locus intersects even a portion of a node, the detection probability previously assigned to that node is set equal to zero. The full value for the detection probability is utilized whenever the entire node lies outside the response time locus. An alternative detection type is "L," for which, in an apportionment similar to that used for intersected arcs, the node detection probability is set equal to a fraction of the assigned probability. This fraction is the percentage of the node (traversal time) penetrated by the adversary in order to reach the locus. Thus the node detection probability is a linear function of node penetration. These distinctions will be made clearer in the discussion of the subroutine MNDPT1 and Figure 7 (p. 45).

All arc detection probabilities are computed from the arc traversal times and, in order to introduce the prior probability distributions into the computational scheme, XPATH calls DFLT.

Subroutine DFLT (Distributions) -- The user may assign any of five different probability distributions to the node and arc traversal and guard response times. All node and arc detection probabilities are assigned the deterministic distribution by DFLT. Five prior distributions are available for the time parameters:

- Deterministic (1)
- Uniform (2)
- Triangular (3)
- Normal (4)
- Lognormal (5)

The user makes the assignment by specifying the appropriate code numbers (1 through 5) for the node traversal time, arc traversal time, and response time. Any permutation of three numbers is permissible (3,2,1; 1,2,3; 4,5,2; etc.) when interruption probability is the measure. With time the measure, the response time distribution is automatically made deterministic and therefore only code numbers associated with node and arc traversal time can be permuted.

DFLT regards a value for an input time parameter (e.g., node or arc traversal time, guard response time) as the mode (most probable value) for the distribution which is assigned to the parameter. Most distributions are conditioned by DFLT so that they are symmetrical, and therefore the mean or expected value equals the mode. For the lognormal distribution, the mean is unequal to the mode. It is, in fact, 1.06 times the mode as a result of setting the coefficient of variation (standard deviation/mean) equal to 0.20. Indeed, with the exception of the uniform distribution, all distributions have coefficients of variation that equal or are set equal to 0.20 by DFLT (uniform, ~0.3) Given the distributions and the modes, DFLT computes values of distribution-dependent quantities for use in the Phase 1 and Phase 2 computations. Appendix F describes the distributions and provides expressions for these quantities (minimum, maximum, mean, standard deviation) that characterize the distributions. In Phase 1 and Phase 2 computations that use Monte Carlo simulation of the distributions (MNDPT1 and EZ), random values for the time parameters are obtained in terms of these characterizing quantities and a randomly generated number, RN, where $0 \leq RN \leq 1$. Phase 2 analytic evaluations (THYME) use the modes directly for those distributions for which the means equal the modes; otherwise, computed means are used.

If, at this point, the user wishes to modify or simply to examine the distributions and the values of the quantities that characterize them, the PATHS procedure calls MOD.

Subroutine MOD (Type Codes) -- The user may examine or modify the distributions and the values of the quantities associated with the five types of input parameters according to six type codes (shown in parentheses); these are:

- Terminate modifications (0)
- Node traversal time (1)
- Node detection probability (2)
- Arc traversal time (3)
- Arc detection probability (4)
- Terminal (target) node response time (5)

There are, however, a number of constraints placed on the use of these type codes. Although values may be changed, the distributions for the node and arc detection probabilities will remain deterministic. If the measure is time, neither the response time distributions (deterministic) nor response time values (mode = BIG) can be changed. Otherwise, both the values and distributions associated with the time parameters can be altered. For each type code specified, MOD displays the current distribution and the current values for the applicable characterizing quantities (mode, min, max, mean, standard deviation) that were previously determined either by input or during the conditioning of the input data by DFLT. PATHS then prompts the user to specify a distribution (new or unchanged) and a new set of characterizing values. If the distribution is to be unchanged, the user specifies a "0" (0 = null distribution). Where necessary (lognormal distribution only), MOD conditions these values for use in MNDPT1 and EZ in Phases 1 and 2. One of the results of this conditioning is that the coefficient of variation for the lognormal distribution remains equal to 0.20. For other distributions, the coefficient of variation will depend upon the new characterizing values that are specified.

Modifications may be terminated by entering the "0" type code. Whether changes have been carried out or not, XPATH transfers control to PATHS to begin Phase 1 operations.

Subroutine PATHS (Storing Information) -- PATHS first initializes the working storage to be used in Phase 1. Throughout the Phase 1 runs, PATHS will collect and store path and node information that is generated for each run. A distinction must be made with regard to information that is saved and information that is dumped. When saved, all information generated is stored, when dumped, only a portion of the information generated is stored. A generated path may be very long; that is, it may exceed the linear storage limit that has been set for a path description (nodal sequence from target to boundary), or more than 29 distinct paths may have been generated thereby exceeding the maximum number of descriptions that can be stored. As long as the limits have not been exceeded during a simulation, PATHS saves and stores a description or nodal sequence for each path generated, a path and node count (i.e., the number of times each node and each path is generated in MRUN runs), and the conditional interruption probability or conditional traversal time for each generated path. When the limits are exceeded and all information that can be saved is saved and stored, PATHS dumps a portion of the excess by storing (1) a dump count or the number of times a dump is made, (2) the aggregate of the conditional interruption probabilities or conditional traversal times for all the generated paths that have exceeded the limits, and (3) a count for each node along the path each time the path is dumped.

Contributions from the node counts in the dump are later added to the node counts that have already been saved to give complete node ranking information for the output. The fractional dump count, DUMP/MRUN, is printed in the output summary as the percentage of paths that are dumped but not saved and this fractional count is also displayed as the rank of the aggregate conditional interruption probability or conditional traversal time mass in the dump. Estimates of the mean and standard deviation of this mass are displayed with the rank. Because, with the exception of the node count, the dump is treated as an aggregate, no Phase 2 evaluation for the dump would be meaningful and indeed the upper, as well as the lower bound, applies only to paths that have been saved.

If the measure is time, the subroutine PATHS will call MNDPT1 directly. If the measure is interruption, PATHS will first call PDIJK before proceeding to MNDPT1.

Subroutine PDIJK (Detection Probabilities) -- PDIJK searches for and identifies least detection probability paths from the facility boundaries to every node in the facility graph. As a result, each node is at some time regarded as a terminal node (end of path) and the conditional detection probability or minimum cumulative probability of detection up to but not including this node is computed and stored. Predecessors (preceding nodes) of each node along the minimum detection paths are also recorded so that these paths can be traced and described.

The cumulative probability of detection (CPD) for a path from a boundary up to but not including terminal node n+1 is P_{n+1}. P_{n+1} is often expressed as the ring sum of all detection probability weightings for the components (nodes and arcs) along the path; that is,

$$P_{n+1} = d_1 \oplus d_{1,2} \oplus d_2 \oplus \ldots \oplus d_n \oplus d_{n,n+1}$$

where d_i are the detection probability weightings for nodes and $d_{i,j}$ are those for arcs. Hence,

$$P_{n+1} = d_1 + (1-d_1)d_{1,2} + (1-d_1)(1-d_{1,2})d_2 + \ldots$$

by definition of the ring sum.

In general (see Appendix G), P_{n+1} is given by

$$P_{n+1} = \sum_{i=1}^{n} d_i \prod_{j=0}^{i-1} (1-d_j)$$

where the sum is over the n path components up to node n+1. Here $d_0 = 0$ and n is an even number that refers to the arc preceding terminal node n+1. (Double subscripts have been eliminated by letting odd i refer to nodes and even i refer to arcs.) In order to compute the CPDs, however, PDIJK makes repeated use of the recurrence relation

$$P_i = P_j + (1-P_j)d_j + (1-P_j)(1-d_j)d_{j,i}$$

that is, the CPD for node i is obtained from the CPD for its predecessor node (node j), the detection probability weighting for node j, d_j, and the detection probability weighting for the arc joining nodes i and j, $d_{j,i}$. The facility nodes are considered as being numbered from 1 to n,* but the numbering is such that j is greater than i; that is, indices are smaller for inner nodes than for outer nodes.

Initially, the P_i for boundary nodes are set equal to zero and the P_i for all other nodes are set equal to BIG (a very large number). PDIJK focuses first on the boundary node j = n (which has no predecessor) and computes the P_i starting at i = 1, for every node i ≠ j that is directly joined to predecessor node n. The P_i for every node not joined to node n retain their initial values (zero or BIG). All the n-1 P_i are compared and PDIJK makes note of the node i = k < n for which P_i is a minimum and for which i is least (in case two or more P_i are minima and equal). This completes the first cycle. On the second cycle, PDIJK sets j = k and computes the P_i, starting again at i = 1, for every node $i \neq k$ that is directly joined to predecessor k. If, for some node i joined to k, a new value of P_i is obtained that is less than the initial value or less than the value computed on a previous cycle, then the old value is replaced by the new and the old predecessor is replaced by a new one; otherwise, the old value and predecessor are retained. The P_i for nodes not joined to node k also retain their old values (and predecessors). The n-2 P_i are compared and PDIJK again notes the node for which P_i is a minimum and begins another cycle of computations and comparisons with j equal to the index of this node. This cycling process results in progress inward from the boundary and continues until no more comparisons can be made. At the completion of PDIJK, the resulting P_i are minimum CPDs up to but not including node i and the minimum detection path from the boundary up to node i may be found by tracing the succession of stored predecessors.

---

*These indices are used for bookkeeping and are not labels (names of the nodes).

The distribution assigned to the node and arc detection probabilities is deterministic and the values for these weights are fixed for each simulation. Consequently, subroutine PDIJK is carried out only once for each simulation. The minimum CPDs computed by PDIJK are then used by MNDPT1 to compute conditional interruption probabilities but, because random sampling of the time parameters is introduced as well, MNDPT1 requires MRUN realizations of these parameters for each simulation.

Subroutine MNDPT1 (Monte Carlo Simulation) -- Given the user specified target nodes, corresponding guard response times, and response time distributions, MNDPT1 locates the response time locus about each target node with respect to the other facility nodes on each of the Phase 1 runs. If the measure is time, the response time distributions are deterministic, the response time for each target is BIG, and the locations of the response time loci are, in effect, immediately known. All facility nodes are located inside these loci, and there exists a minimum time path from each target to the boundary of the facility. In this case then, MNDPT1 searches the facility graph, which is weighted only by node and arc traversal times, and finds that path (and target) for which the minimum traversal time is least on each of the MRUN realizations of the traversal time parameters. When interruption is the measure, MNDPT1 first searches the time weighted graph and finds minimum time paths from each target up to and including the nodes and arcs or portions thereof that lie on the response time locus associated with each target. From each response time locus to the facility boundary the graph is weighted by detection probabilities and there exists a path of minimum interruption probability, from each target to the boundary that combines a segment of the path that minimizes traversal time from the target to the locus with a segment of the path that minimizes detection probability from the boundary to the locus. MNDPT1 then finds that path (and target) for which the interruption probability is least on each of the MRUN realizations of the time (response and delay time) parameters.

Each realization consists of a new set of values comprising the values of the traversal times for every arc and node and, when the measure is interruption, the value of the guard response time for each and every specified target. Given the values that characterize the prior probability distributions (from DFLT and MOD), this set is generated by Monte Carlo sampling of these distributions that are assigned to the time parameters. Appendix H describes the Monte Carlo sampling method that is used not only by MNDPT1 in the path generation phase but also by EZ in the path evaluation phase.

Because the Dijkstra-Yen search is used in both MNDPT1 and PDIJK, the search methods used by both are similar in all important respects. The method used in MNDPT1 differs from that used in PDIJK only by the direction in which it is carried out; in MNDPT1, the search progresses outward from the targets rather than inward from the boundary.

The cumulative traversal time for a path from a target up to and including component j (or a portion thereof) is simply the sum of the time weightings for the components (nodes and arcs) or

$$T_j = \sum_{i=1}^{j-1} t_i + f_j t_j$$

where $0 \leq f_j \leq 1$. Hence, repeated use is made of the recurrence relation

$$T_j = T_i + f_j t_j$$

The numbering of the indices is unchanged from that used in PDIJK.

For each run, all user specified targets are considered. As each target is treated in turn, the T_j for all nodes but the target are set equal to BIG; T_j for the target is its traversal time. MNDPT1 first focuses on target node i=m,* sets the predecessor of node m equal to zero (none), and computes the T_j, beginning with j=n, for each node j that is directly joined to predecessor node m. The n-1 T_j are compared and the node j=k for which T_j was a minimum is noted. If T_k for this node is less than the response time, RT(m), MNDPT1 sets the predecessor of node k equal to m, sets i=k, and then, on the second cycle, computes the T_j starting again at j=n, for each node j that is directly connected to predecessor k. The values of the T_j and the predecessors of nodes j are revised or retained as in PDIJK. The resulting n-2 T_j are compared, and the minimum T_j and its associated node are noted. This cycling process continues and progress is made outward from the target until the rth cycle occurs in which a minimum T_j equals or exceeds RT(m). All n-r nodes for which the T_j are not equal to BIG are locus nodes. The positions of points (locus points) that are exactly a response time away from a given target m are then fixed with respect to the corresponding locus nodes by requiring that each T_j = RT(m). By setting the T_j = RT(m), the values of the f_j are determined; these values indicate the fractional portions of locus nodes or arcs from locus nodes toward the target that are intersected by the locus itself at the locus points. The minimum cumulative detection probabilities up to but not including the locus nodes j are already known from PDIJK. Once the values for $(1-f_j)$ have been obtained, the detection probability contributions from the locus nodes and from the locus nodes up to the locus points can be computed. The minimum CPD that is computed is the minimum CPD for the target under consideration. As MNDPT1 computes minimum CPDs for all targets during a given run, comparisons of these values are made and the least value that results for one of these targets is the minimum interruption probability or conditional probability of interruption for the path that extends from that target to the boundary.** Figure 7 is a simple facility graph that may be used to show how conditional interruption probabilities, the P_c, are obtained for two different runs.

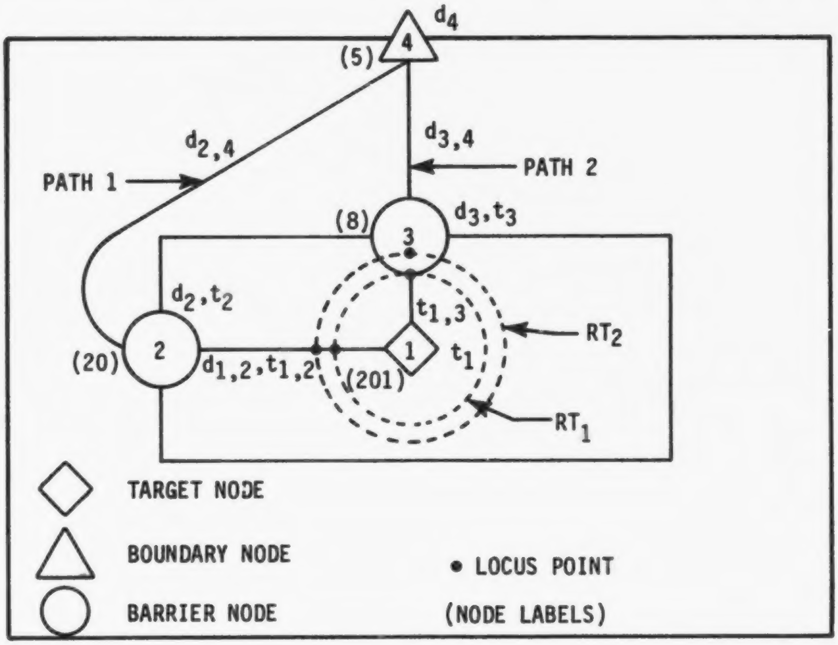
---

*This number is an index, not a label

**Because PDIJK computes the CPDs for each node without regard to the location of any targets, it is possible that a CPD for a particular locus node includes contributions from least detection path segments that first enter the response time locus from another locus node and subsequently exit at the node in question. Such a path can never be selected as a minimum interruption path because there exists a path with a smaller CPD that proceeds directly from the entrance node to the target.

The facility graph in Figure 7 consists of 4 nodes and 4 arcs, i.e., one boundary node, two barrier nodes, one target node, and the arcs joining them. The nodes (labels in parentheses) are indexed from 1 to 4, starting at the single target. Although detection probability and traversal time weights are associated with each node and arc, only those weights that are essential to the computations are shown. Two different runs, corresponding to response times RT = RT_1 and RT = RT_2 (designated by dashed circles), are used to show the effect of node detection type (NTYP = L or S) upon the computations. Nodes 2 and 3 are locus nodes. For the run on which RT = RT_1, node detection type is inconsequential because no node is intersected by the response time locus. For this case then, the interruption probabilities at the locus points for paths 1 and 2 are given under (1) and (2) of Figure 7. Note that these probabilities are expressed in terms of the minimum CPDs obtained in PDIJK. For the run on which RT = RT_2, locus node 3 is intersected by the response time locus. Consequently, when NTYP = L, a fraction (1-1/4 = 3/4 in Figure 7) of the locus node probability d_3 is contributed to the minimum CPD along path 2, but when NTYP = S, fractional contributions do not matter and d_3 is set equal to zero. The interruption probabilities for RT = RT_2 are given under (1) and (3) of Figure 7 for paths 1 and 2. The least value for the interruption probability is the conditional probability of interruption, P_c, for the run and this value, along with the path description (201, 20, 5 or 201, 8, 5), is what subroutine PATHS collects before recalling MNDPT1 for the next Phase 1 run.

Subroutine PATHS (Collection of Phase 1 Information) -- When MNDPT1 completes a run, a return to subroutine PATHS is made and information generated by the run is either saved or dumped. Subroutine PATHS then recalls MNDPT1 for the next run. This procedure is repeated until all MRUN Phase 1 replications have been carried out. When the path generation phase is complete, subroutine PATHS will have saved the sum and the sum of the squares of the conditional probabilities of interruption (or conditional traversal times) over all runs for (1) each distinct path and for (2) all the generated paths. The sum and the sum of the squares for each distinct path (along with the count for the path) may be used in XPATH to compute Phase 1 sample statistics for the path; namely, the arithmetic mean and standard deviation of the sample that comprises



(1) Interruption probability for path 1 where $RT = RT_1$ or RT_2 and $NTYP = L$ or S:

$$P = P_2 + (1-P_2)d_2 + (1-P_2)(1-d_2)d_{1,2}(1-f_{1,2}), \text{ where}$$

$$P_2 = d_4 + (1-d_4)d_{2,4} \text{ from PDIJK, and}$$

$$(1-f_{1,2}) = (T_{1,2}-RT)/t_{1,2} = (t_1+t_{1,2}-RT)/t_{1,2}$$

(2) Interruption probability for path 2 where $RT = RT_1$ and $NTYP = L$ or S:

$$P = P_3 + (1-P_3)d_3, \text{ where}$$

$$P_3 = d_4 + (1-d_4)d_{3,4} \text{ from PDIJK}$$

(3) Interruption probability for path 2 where $RT = RT_2$:

$$P = P_3 + (1-P_3)d_3(1-f_3) \quad \text{for NTYP = L, and}$$

$$P = P_3 \quad \text{for NTYP = S, where}$$

$$P_3 = d_4 + (1-d_4)d_{3,4} \text{ from PDIJK, and}$$

$$(1-f_3) = (T_3-RT)/t_3 = (t_1+t_{1,3}+t_3-RT)/t_3$$

Figure 7. Facility Graph and Interruption Probabilities for Two Runs.

those replications that generate the path. These statistics are estimates of the mean conditional interruption probability (or traversal time) and square root of the variance (sample variance) of the conditional interruption probabilities (or traversal times) that were computed for each path. The sum and the sum of the squares over all paths (along with the specified value for MRUN) will be used in XPATH to compute Phase 1 sample statistics for the facility; namely, the arithmetic mean and standard deviation of the sample that comprises all Phase 1 replications. These statistics are the lower bound and square root of the Phase 1 sample variance.

From subroutine PATHS, control is returned to XPATH in which the course of the path evaluation (Phase 2) treatment is determined. If the specified measure is time, control passes to THYME; if the measure is interruption (detection), control is given to EZ.

Subroutines THYME and EZ (Path Evaluation) -- For the Phase 2 path evaluation, all paths generated in Phase 1 are examined unconditionally, i.e., without regard to minimization of the measure. The specified measure indicates whether the random variable is traversal time or interruption probability and hence whether the evaluation can be carried out by analytic or Monte Carlo methods. With time the measure, the random variable, T, is a function of only the node and arc traversal times, and the expected value and variance of the traversal time for a given path is simply the sum of the expected values and variances of the n path components; that is,

$$\begin{aligned} E_T(T) &= \int_{-\infty}^{\infty} \sum_{i=1}^{n} t_i f(t_i) dt_i \\ &= \sum_{i=1}^{n} \int_{-\infty}^{\infty} t_i f(t_i) dt_i \\ &= \sum_{i=1}^{n} E_T(t_i) \end{aligned}$$

and

$$Var_T(T) = \sum_{i=1}^{n} Var_T(t_i)$$

where the t_i are the component traversal times and the $f(t_i)$ are the prior probability distributions for these traversal times. Consequently, the subroutine THYME computes the sum of the $E_T(t_i)$ and the sum of the $Var_T(t_i)$ along each path using the means and variances (σ^2) given analytically by the expressions in Appendix F for the applicable prior probability distributions. The resulting statistics are Phase 2 criticality estimates (estimates of mean unconditional traversal times) and estimates of precision (standard deviations) for each path. XPATH will determine the value of the Phase 2 facility estimate, the upper bound MIN, by finding the minimum in the set of all estimates of the mean unconditional traversal times.

When interruption probability is the measure, the random variable, $P_u = P[E(D),T]$, is a function of the expected values for the detection probabilities and the delay and response times. Even though the paths are known and the expected values for the detection probabilities are fixed, the Monte Carlo approach is used because the analytic evaluation, for example, of

$$E_T(P_u) = \int_{-\infty}^{\infty} P_u f_2(t)dt$$

where $f_2(t)$ is the marginal density function for the delay and reponse times, is not possible. Therefore, for each distinct path, subroutine EZ computes P_u IRUN times; i.e., for IRUN realizations of the response time and component traversal times. On each run, subroutine EZ uses Monte Carlo sampling of the prior probability distributions for the response time and node and arc traversal times that are associated with a known path. The response time locus is determined by computing the cumulative traversal time along the time segment of the path from the known target to the known locus node and then establishing the locus point as was done in MNDPT1. The CPD for the known detection segment that extends from the locus point to the boundary (equal to the CPD from the boundary to the locus point) is then computed using the fixed detection weights and the same recurrence relation that was used in PDIJK. A running sum and sum of the squares of the P_u are kept during the IRUN realizations for each distinct path. These sums are used in XPATH (along with the specified value of IRUN) to compute the sample mean (arithmetic mean) and sample standard deviation for each path. These statistics are Phase 2 criticality estimates (estimates of mean unconditional probabilities of interruption) and estimates of precision for each path. XPATH will determine the value of MIN, the least of these estimates of the mean unconditional probabilities of interruption.

XPATH (Computation of Statistics and Output) -- When the evaluation phase is complete, control is transferred back to XPATH from EZ or THYME. Except for determining the total counts for each node and identifying the path for which the Phase 2 mean unconditional interruption probability or traversal time is a minimum, all information essential to the computation of the Phase 1 and Phase 2 statistics (estimates) and to the printing of the output is complete at this stage of the procedure. Therefore the total count, equal to the sum of the counts that have been saved and dumped, is computed for each node that is an element of a path that was generated in Phase 1, and the sums of the P_u or $E(t_i)$ that were computed for each path in EZ or THYME are compared to identify the Phase 2 minimum path. Knowing this path, the statistic, which is the estimate, MIN (or upper bound), can be computed from the appropriate sum, and the corresponding estimates of precision can be computed using the sum of the squares of the P_u or the sum of the $Var(t_i)$. As all Phase 1 and Phase 2 statistics are computed, XPATH prints the output of a PATHS simulation under four headings, namely: PATHS Summary, Path Description, Path Ranking/Interruption Information or Path Ranking/Time Information, and Node Ranking Information.

The "PATHS Summary" provides (1) user-specified information such as the title of the simulation, the random number seed, the number (MRUN) of pathfinding (Phase 1) replications, and the number (IRUN) of interruption (Phase 2) replications/path saved and (2), information generated by the PATHS procedure such as the number of distinct paths saved, the percentage (DUMP/MRUN) of paths found (generated) but not saved, and the Phase 1 and Phase 2 facility estimates (mean of the Phase 1 sample and least mean of the Phase 2 samples) or the lower and upper bounds along with the corresponding estimates of precision (sample standard deviations, standard deviations of the sample means, and the coefficient of variation).* The "Path Description" consists of a listing of the node labels for each generated path in a target-to-boundary sequence. Each path is numbered and the "Path Ranking/Interruption Information" for each path can be obtained with standard or extended content. The extended output contains a listing for each path that includes the path number, the path rank (Phase 1 criticality estimate), the cumulative ranking, the estimates of mean conditional (given that the path _is_ the minimum path) interruption probability and standard deviation, the estimates of mean unconditional interruption probability (Phase 2 criticality estimate) and standard deviation, and the estimate of mean conditional (given that the path _is_ _not_ the minimum path) interruption probability.* If the dump is not empty, the dump rank (DUMP/MRUN), the cumulative rank for all paths (both saved and dumped), and the estimates of the mean and standard deviation for the aggregate conditional interruption probability mass are printed also. In addition, the linear correlation coefficient expressing the linear correlation of the Phase 1 and Phase 2 criticality estimates for all _saved_ paths is printed. The standard output includes path numbers, path ranks, cumulative rankings, and unconditional estimates for saved paths as well as the dump information. The "Path Ranking/Time Information" with standard or extended content contains outputs that correspond to those provided with standard or extended content under the "Path Ranking/Interruption Information" heading. (Substitute "traversal time" for "interruption probability" in the sentences above.) For both measures (time and interruption), the "Node Ranking Information" provides a listing by node type (target, boundary, barrier) of every node label and corresponding node rank (node count/MRUN) for nodes that are elements of generated paths (both saved and dumped) in decreasing order of rank.

---

*With time the measure, only the facility estimate and a precision estimate pertaining to Phase 2 are computed and printed in the summary, and these are limited to the least sample mean, MIN, of Phase 2 and the corresponding sample standard deviation. The ratio of the lower to upper bound, LB/MIN, is printed merely to provide a check on the Phase 1 sample size.

In order to provide the output described above, XPATH computes the statistics that constitute the facility estimates, the path criticality estimates, and the estimates of precision. In addition, XPATH computes the ranks of the nodes and the linear correlation coefficient.

---

*The conditional estimate (_is_ _not_ minimum), called NOT MIN in the output, is to be differentiated from the other conditional estimate (_is_ minimum). NOT MIN is _derived_ from the expression given in the discussion of "Path Criticality Estimates" in Section 2, p. 27.

A sample statistic that is an estimate of the population mean is computed as the arithmetic mean of the sampled values. If S_i is a sample value (such as a value for P_c or P_u), then the sample mean, $\underline{S}$, is

$$\underline{S} = \sum_{i=1}^{N} S_i/N$$

where N is the sample size. N can, of course, equal the Phase 1 sample size, MRUN, the Phase 2 sample size, IRUN, or equal the path count (PCNT = number of times generated) for a particular path. The expected value of $\underline{S}$ is $E(\underline{S})$ and

$$E(\underline{S}) = \mu$$

where μ is the mean of the population from which the samples are drawn (with replacement). The standard deviation of the sample, $\underline{V}$, is computed by

$$\underline{V} = \left[\left(\sum_{i=1}^{N} S_i^2 - N\underline{S}^2\right)/(N - 1)\right]^{1/2}$$

The expectation of the variance, $\underline{V}^2$, is $E(\underline{V}^2)$ and

$$E(\underline{V}^2) = \sigma^2$$

that is, the expected value of the unbiased sample variance is the population variance, σ^2.* The standard deviation of the sample mean is given by $\underline{V}/N^{1/2}$ where

$$E(\underline{V}^2)/N = \sigma^2/N$$

and the coefficient of variation is given by $\underline{V}/\underline{S}$ where

$$E(\underline{V}^2)^{1/2}/E(\underline{S}) = \sigma/\mu$$

Simply stated, the statistics $\underline{S}, \underline{V}, \underline{V}/N^{1/2}$, and $\underline{V}/\underline{S}$ that are computed by XPATH can be expected to be good estimates of the population parameters μ, σ, $\sigma/N^{1/2}$, and σ/μ if the sample sizes are sufficiently large.

---

*In general, $E(\underline{V}) \neq \sigma$, and therefore the standard deviation is not unbiased.

The path and node ranks are computed by dividing the appropriate count (PCNT or NCNT) by the Phase 1 sample size (MRUN). The linear correlation coefficient, COR, is computed from

$$\mathrm{COR} = -\ k\left[\sum_{i=1}^{k} r_i\underline{S}_i - S\sum_{i=1}^{k} r_i\right] \Big/ \left[\left(k\sum_{i=1}^{k} r_i^2 - 1\right)\left(k\sum_{i=1}^{k} \underline{S}_i^2 - S^2\right)\right]^{1/2}$$

where

k = number of distinct paths generated and saved,

$\underline{S}_i = E_T(P_u)_i$ or $E_T(T)_i$, the Phase 2 criticality estimate for the ith path,

$S = \sum_{i=1}^{k} \underline{S}_i$

$r_i = \mathrm{PCNT}_i/\mathrm{MRUN}$, the rank or Phase 1 criticality estimate for the ith path,

and the first minus sign in the expression for COR is used in order to make positive the linear correlation between high values for the Phase 1 estimates and low values for the Phase 2 estimates. In this way, the probability that a path is most critical and the probability that it is not interrupted is viewed as a positive correlation.

BLANK PAGE

4. EXECUTION OF PATHS

Input to PATHS

Input to PATHS consists of (1) the facility graph input, including node and arc labels and weights, all of which is stored on an input file, Tape 20, and (2) the input that must be specified by the user at the beginning and during a PATHS simulation. The discussion that follows is confined to the second type of input, which is requested from the user during the course of interaction between the computer system and the user.

Accessing the PATHS Procedure File -- After dialing the number of the Sandia Network Operating System (NOS), the user should wait for the NOS signal (a continuous tone), then attach the phone to the activated terminal and press the "RETURN" key on the terminal. The system will respond with 6000 SYSTEM UP or, if problems have occurred, with 6000 SYSTEM DOWN. If the system is up, it will display

```
80/06/13  09.30.42
SANDIA NOS - 485 SN53        NOS 1,3  485/485
USER NUMBER, PASSWORD
```

The first line is the current date and time. The user must type in the user number, a comma, and the password, and then depress the RETURN key.* The NOS will not echo this input but, if an error has been made, it will display

```
IMPROPER LOG IN, TRY AGAIN.
USER NUMBER, PASSWORD
```

When the user number and password are accepted, the system responds with

---

*All user entries must be ended with a "RETURN."

```
TERMINAL:            34, TTY
RECOVER/SYSTEM:
```

and the user should first type GET,UPATH/UN=DENGI.* When the system displays READY, the user should then enter CALL, UPATH. The terminal number (34 in this example) is to be noted in case the system goes down. If this happens, the user can attempt recovery at his terminal with the response RECOVER/ SYSTEM: RECOVER, 34. If recovery is possible (it is not possible if the user tries to recover more than 15 minutes after the system has come on-line again), the system responds with RECOVERY COMPLETE and instructions on how to proceed. Alternatively, the user may choose to restart the interrupted simulation at the beginning, particularly if the interruption occurs at the beginning of a simulation or if no extensive modifications have been made to the default data that are obtained from Tape 20. In this circumstance, the user will, as in normal operation, enter CALL, UPATH, and the system will respond with PATHS INPUT indicating that the PATHS procedure file has been accessed.

User-Specified Input -- Immediately following the PATHS response, PATHS INPUT, the first in the series of user-specified inputs will be requested. Figures 8, 9, 10, and 11 provide examples of the inputs that must be specified by the user up to the point at which output begins. The four examples include

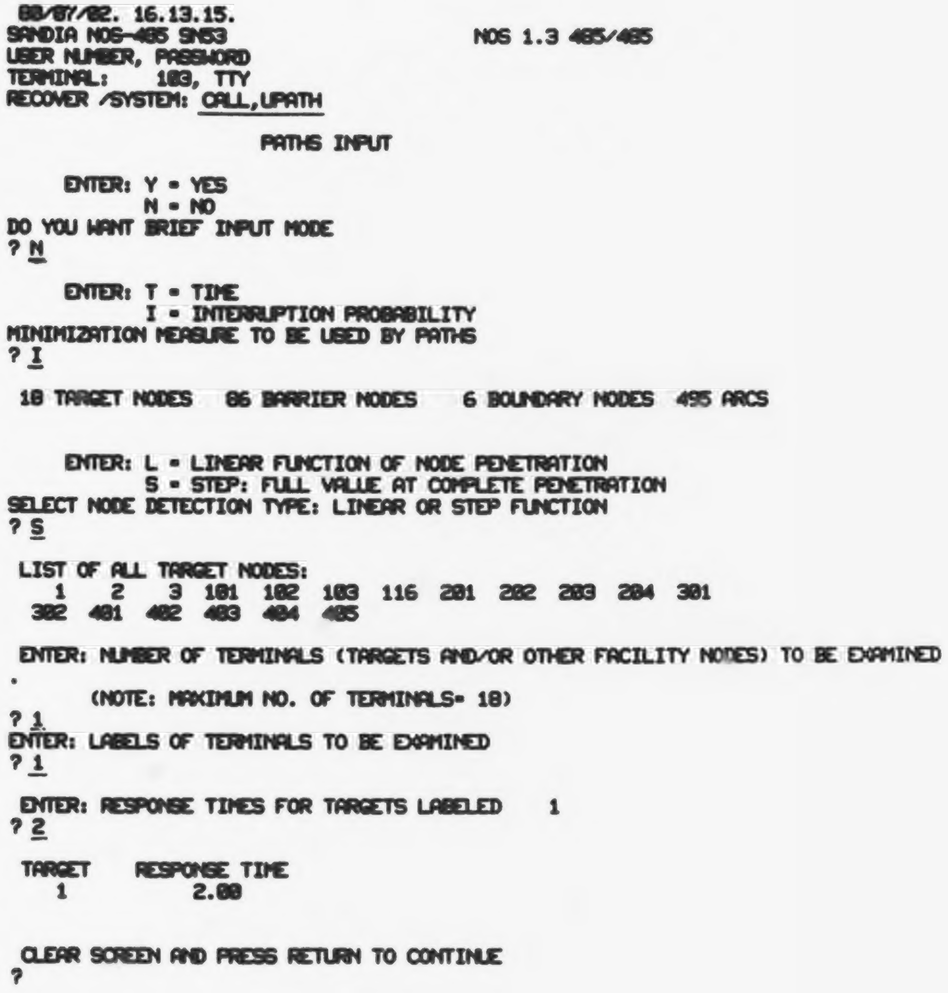
- Figure 8--extended input mode, measure = interruption
- Figure 9--brief input mode, measure = interruption
- Figure 10--extended input mode, measure = time
- Figure 11--brief input mode, measure = time

In addition, Figures 8 and 10 illustrate how modifications may be carried out in the extended mode. (Were the brief mode in use, the method of modification would be the same, but no listing of the type codes or distribution codes would be provided as a guide.)

Figure 8 begins by recapitulating the sign-on and access procedure. The first query in the PATHS procedure requests the type of input mode. The question mark is the prompt for the user input in this procedure. The user response indicates that the extended mode will be used.

---

*User responses are underlined.



```
 08/07/82. 16.13.15.
SANDIA NOS-485 SN53                    NOS 1.3 485/485
USER NUMBER, PASSWORD
TERMINAL:     103, TTY
RECOVER /SYSTEM: CALL,UPATH

                        PATHS INPUT

     ENTER: Y = YES
            N = NO
DO YOU WANT BRIEF INPUT MODE
? N

     ENTER: T = TIME
            I = INTERRUPTION PROBABILITY
MINIMIZATION MEASURE TO BE USED BY PATHS
? I

 18 TARGET NODES   86 BARRIER NODES    6 BOUNDARY NODES  495 ARCS


     ENTER: L = LINEAR FUNCTION OF NODE PENETRATION
            S = STEP: FULL VALUE AT COMPLETE PENETRATION
SELECT NODE DETECTION TYPE: LINEAR OR STEP FUNCTION
? S

 LIST OF ALL TARGET NODES:
    1    2    3  101  102  103  116  201  202  203  204  301
  302  401  402  403  404  405

 ENTER: NUMBER OF TERMINALS (TARGETS AND/OR OTHER FACILITY NODES) TO BE EXAMINED
.
        (NOTE: MAXIMUM NO. OF TERMINALS= 18)
? 1
ENTER: LABELS OF TERMINALS TO BE EXAMINED
? 1

 ENTER: RESPONSE TIMES FOR TARGETS LABELED     1
? 2

 TARGET    RESPONSE TIME
    1           2.00


 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?
```

Figure 8. PATHS Input for an Interruption Simulation Using the Extended Mode

```
ENTER: SEED FOR THE RANDOM NUMBER GENERATOR
? 4621

ENTER: NUMBER OF PATHFINDING REPLICATIONS DESIRED
? 117
ENTER: NUMBER OF INTERRUPTION CALCULATION REPLICATIONS/PATH SAVED
? 1000

         DISTRIBUTION CODE

     0 = NULL
     1 = DETERMINISTIC
     2 = UNIFORM
     3 = TRIANGULAR
     4 = NORMAL
     5 = LOGNORMAL

ENTER: DISTRIBUTION CODE FOR NODE TRAVERSAL TIMES
? 3
ENTER: DISTRIBUTION CODE FOR ARC TRAVERSAL TIMES
? 3
ENTER: DISTRIBUTION CODE FOR RESPONSE TIMES
? 3

     ENTER - Y = YES
             N = NO
DO YOU WANT TO MODIFY OR EXAMINE ANY DISTRIBUTIONS
? Y

              TYPE CODE

     0 = TERMINATE MODIFICATIONS
     1 = NODE TRAVERSAL TIME
     2 = NODE DETECTION PROBABILITY
     3 = ARC TRAVERSAL TIME
     4 = ARC DETECTION PROBABILITY
     5 = TERMINAL NODE RESPONSE TIME


ENTER: TYPE CODE
? 2
ENTER: NODE LABEL
? 1

        NODE    1 DETECTION PROBABILITY
        DETERMINISTIC VALUE =   0.0000

ENTER: DISTRIBUTION CODE
? 3
ENTER: DETERMINISTIC VALUE
? 1
```

Figure 8. PATHS Input for an Interruption Simulation Using the Extended Mode (Continued)

```
ENTER: TYPE CODE
? 2
ENTER: NODE LABEL
? 1

        NODE    1 DETECTION PROBABILITY
        DETERMINISTIC VALUE =   1.0000

ENTER: DISTRIBUTION CODE
? 0

ENTER: TYPE CODE
? 3
ENTER: NODE LABEL PAIR
? 1,20

        ARC (   1,  20) TRAVERSAL TIME
        TRIANGULAR   MODE =    .1114   MIN =    .0557   MAX =    .1672

ENTER: DISTRIBUTION CODE
? 3
ENTER TRIANGULAR PARAMETERS: MODE MIN MAX
? 1,3,2
MIN MUST BE LESS THAN MODE
AND MODE MUST BE LESS THAN MAX
ENTER TRIANGULAR PARAMETERS: MODE MIN MAX
? 2,3,1
MIN MUST BE LESS THAN MODE
AND MODE MUST BE LESS THAN MAX
ENTER TRIANGULAR PARAMETERS: MODE MIN MAX
? 2,1,3

ENTER: TYPE CODE
? 3
ENTER: NODE LABEL PAIR
? 1,20

        ARC (   1,  20) TRAVERSAL TIME
        TRIANGULAR   MODE =   2.0000   MIN =   1.0000   MAX =   3.0000

ENTER: DISTRIBUTION CODE
? 0

ENTER: TYPE CODE
? 0

ENTER: HEADER
? I EXAMPLE


 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?
```

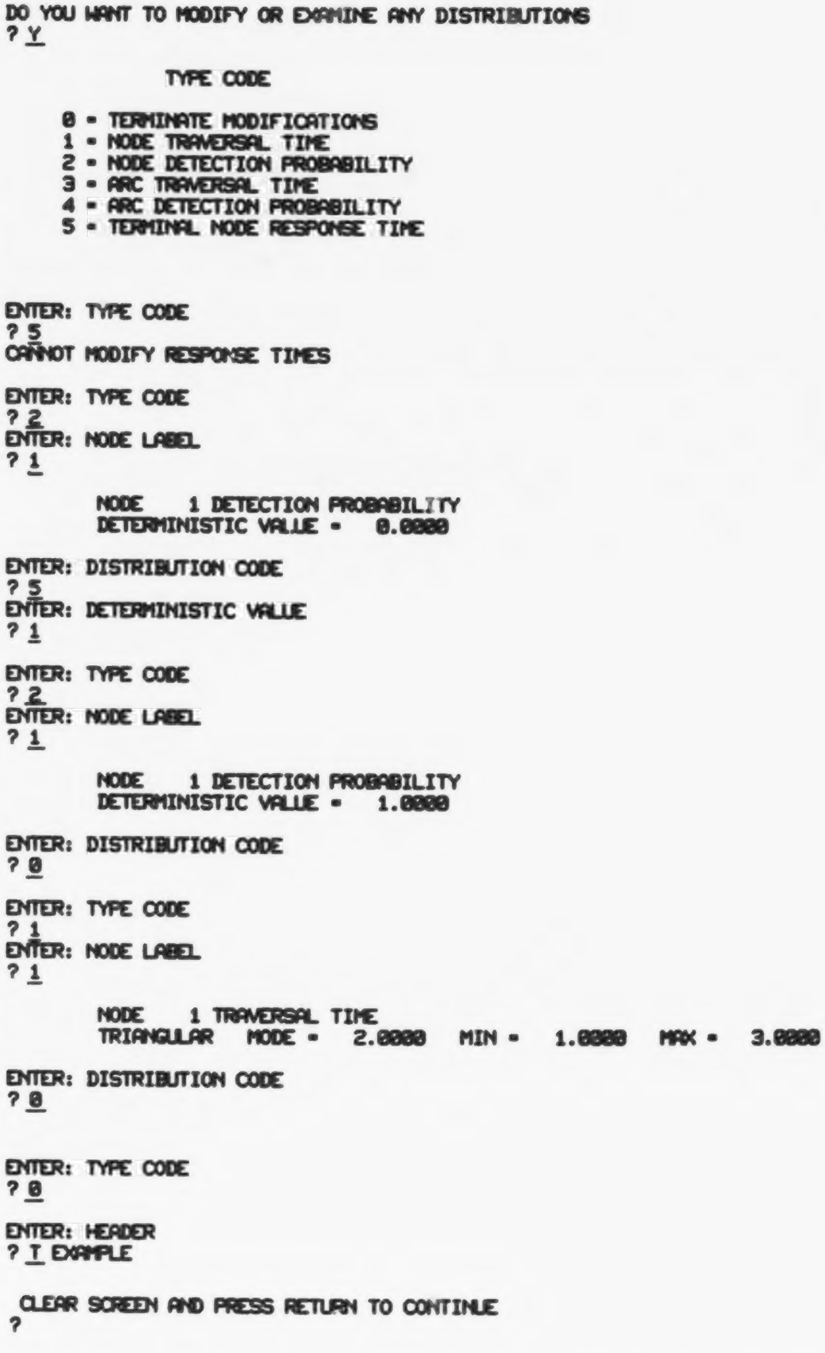
Figure 8. PATHS Input for an Interruption Simulation Using the Extended Mode (Concluded)

```
CALL,UPATH

                        PATHS INPUT

     ENTER: Y = YES
            N = NO
DO YOU WANT BRIEF INPUT MODE
? Y

     ENTER: T = TIME
            I = INTERRUPTION PROBABILITY
MINIMIZATION MEASURE TO BE USED BY PATHS
? I

 18 TARGET NODES   86 BARRIER NODES    6 BOUNDARY NODES  495 ARCS


     ENTER: L = LINEAR FUNCTION OF NODE PENETRATION
            S = STEP: FULL VALUE AT COMPLETE PENETRATION
SELECT NODE DETECTION TYPE: LINEAR OR STEP FUNCTION
? S

 LIST OF ALL TARGET NODES:
    1    2    3  101  102  103  116  201  202  203  204  301
  302  401  402  403  404  405

 ENTER: NUMBER OF TERMINALS (TARGETS AND/OR OTHER FACILITY NODES) TO BE EXAMINED
.
       (NOTE: MAXIMUM NO. OF TERMINALS= 18)
? 1
ENTER: LABELS OF TERMINALS TO BE EXAMINED
? 1

 ENTER: RESPONSE TIMES FOR TARGETS LABELED    1
? 2

 TARGET    RESPONSE TIME
    1           2.00


 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?


ENTER: SEED,MRUN,IRUN, TW,DAWT,DRT
? 4621,117,1000,3,3,3

DO YOU WANT TO MODIFY OR EXAMINE ANY DISTRIBUTIONS
? N


ENTER: HEADER
? I EXAMPLE


 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?
```

Figure 9. PATHS Input for an Interruption Simulation Using the Brief Mode

```
                       PATHS INPUT

     ENTER: Y = YES
            N = NO
DO YOU WANT BRIEF INPUT MODE
? N

     ENTER: T = TIME
            I = INTERRUPTION PROBABILITY
MINIMIZATION MEASURE TO BE USED BY PATHS
? T

 18 TARGET NODES   86 BARRIER NODES    6 BOUNDARY NODES  495 ARCS


 LIST OF ALL TARGET NODES:
    1    2    3  101  102  103  116  201  202  203  204  301
  302  401  402  403  404  405

 ENTER: NUMBER OF TERMINALS (TARGETS AND/OR OTHER FACILITY NODES) TO BE EXAMINED
 .
       (NOTE: MAXIMUM NO. OF TERMINALS= 18)
? 1
ENTER: LABELS OF TERMINALS TO BE EXAMINED
? 1

ENTER: SEED FOR THE RANDOM NUMBER GENERATOR
? 4621

ENTER: NUMBER OF PATHFINDING REPLICATIONS DESIRED
? 117

           DISTRIBUTION CODE

      0 = NULL
      1 = DETERMINISTIC
      2 = UNIFORM
      3 = TRIANGULAR
      4 = NORMAL
      5 = LOGNORMAL

ENTER: DISTRIBUTION CODE FOR NODE TRAVERSAL TIMES
? 3
ENTER: DISTRIBUTION CODE FOR ARC TRAVERSAL TIMES
? 3

      ENTER - Y = YES
              N = NO
```

Figure 10. PATHS Input for a Time Simulation Using the Extended Mode



```
DO YOU WANT TO MODIFY OR EXAMINE ANY DISTRIBUTIONS
? Y

              TYPE CODE

     0 = TERMINATE MODIFICATIONS
     1 = NODE TRAVERSAL TIME
     2 = NODE DETECTION PROBABILITY
     3 = ARC TRAVERSAL TIME
     4 = ARC DETECTION PROBABILITY
     5 = TERMINAL NODE RESPONSE TIME


ENTER: TYPE CODE
? 5
CANNOT MODIFY RESPONSE TIMES

ENTER: TYPE CODE
? 2
ENTER: NODE LABEL
? 1

        NODE    1 DETECTION PROBABILITY
        DETERMINISTIC VALUE =   0.0000

ENTER: DISTRIBUTION CODE
? 5
ENTER: DETERMINISTIC VALUE
? 1

ENTER: TYPE CODE
? 2
ENTER: NODE LABEL
? 1

        NODE    1 DETECTION PROBABILITY
        DETERMINISTIC VALUE =   1.0000

ENTER: DISTRIBUTION CODE
? 0

ENTER: TYPE CODE
? 1
ENTER: NODE LABEL
? 1

        NODE    1 TRAVERSAL TIME
        TRIANGULAR   MODE =   2.0000   MIN =   1.0000   MAX =   3.0000

ENTER: DISTRIBUTION CODE
? 0


ENTER: TYPE CODE
? 0

ENTER: HEADER
? T EXAMPLE

 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?
```

Figure 10. PATHS Input for a Time Simulation Using the Extended Mode (Concluded)

```
                         PATHS INPUT

     ENTER: Y = YES
            N = NO
DO YOU WANT BRIEF INPUT MODE
? Y

     ENTER: T = TIME
            I = INTERRUPTION PROBABILITY
MINIMIZATION MEASURE TO BE USED BY PATHS
? T

 18 TARGET NODES   86 BARRIER NODES    6 BOUNDARY NODES  495 ARCS


 LIST OF ALL TARGET NODES:
    1    2    3  101  102  103  116  201  202  203  204  301
  302  401  402  403  404  405

 ENTER: NUMBER OF TERMINALS (TARGETS AND/OR OTHER FACILITY NODES) TO BE EXAMINED
.
      (NOTE: MAXIMUM NO. OF TERMINALS= 18)
? 1
ENTER: LABELS OF TERMINALS TO BE EXAMINED
? 1

ENTER: SEED,MRUN,DW,DAWT
? 4621,117,3,3

DO YOU WANT TO MODIFY OR EXAMINE ANY DISTRIBUTIONS
? N

ENTER: HEADER
? T EXAMPLE

JOB ACTIVE.


 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?
```

Figure 11. PATHS Input for a Time Simulation Using the Brief Mode

If the user makes an input error such as typing Y when N was intended, two remedies are possible prior to depressing "RETURN." The user can depress the keys "CTRL" and "H" simultaneously and then retype the correct character. "CTRL H" essentially backspaces and erases and, therefore, a character string can be corrected by backspacing to the first error and retyping from that point on. When corrected, the response is entered by a "RETURN." Alternatively, the user can depress the "ESC" key. The system will respond with *DEL*, indicating that all characters following the prompt have been deleted and will advance the page by one line. The response should then be typed anew and entered by the usual "RETURN."

After accepting the measure to be used in this simulation, the PATHS procedure indicates the facility graph size by listing the number of arcs and nodes, and the types of nodes.

Because the measure is interruption in Figure 8, the procedure requests the node detection type (NTYP). If the measure were time as in Figure 10, no such request would be made because detection probabilities are irrelevant under this measure. With interruption the measure, the user can enter S indicating that values for all node detection probabilities are limited to the full assigned values or zero (step function values). When only a portion of a node is intersected by a response time locus, the value of the detection probability for that node is set equal to zero; otherwise, the full value is retained. The alternative detection type is L, indicating that all node detection probabilities may take on continuous values from zero to the full assigned values. For a given intersected node, the value is a linear function of the fraction of that node's traversal time that is covered by the adversary in reaching the response time locus.

Following this response (S or L), PATHS provides a list of the labels for all target nodes. (This information is obtained from a file on Tape 20). For every target to be examined, a label must be supplied by the user.* In this example, the target label is 1. If more than one label is required, they should be listed (1,201,..., etc.) using a comma as a separator (delimiter). The response time (in minutes) for each target is similarly entered (2,4,..., etc.) in the order corresponding to the order of the labels. Note that the response time is treated as a floating point number whether or not it is entered in integer or floating point form.

---

*Only labels that correspond to the listed targets are acceptable. If a barrier or boundary node label is entered, PATHS will reject this input and request that all labels be re-entered.

The request to CLEAR SCREEN AND PRESS RETURN TO CONTINUE is repeated frequently in order to accommodate the display to a CRT screen. Following the prompt, the user presses "RETURN." Note that PATHS does not directly acknowledge this response but continues to the next step in the procedure.

The random numbers generated in PATHS constitute a sequence of pseudo-random numbers between zero and one. The numbers are pseudo-random numbers because they are generated by a completely deterministic method. Nonetheless, the method is such that the statistical properties of these numbers are similar to the conceptual properties of random numbers. The seed is any number* (integer SEED), and it serves as a starting point for the sequence. The same sequences will be generated as long as the seed is the same. Therefore, a simulation can be repeated if all the inputs, including the random number seed, are the same. In the examples of Figures 8 through 11, the seed is 4621 but any other integer could have been chosen. The number of pathfinding replications, MRUN, is 117 and the number of Phase 2 replications per path saved, IRUN, is 1000. Because the extended mode is in use in Figure 8, a listing of the distribution codes is given as a guide in the specification of the prior probability distributions for the node and arc traversal times and the response time(s). Distribution codes for the node and arc detection probabilities need not be given. These distributions are deterministic and, by default, code 1 has been assigned to these probabilities. The triangular distribution assignment has been specified for all time parameters.

At this point, except for the title (HEADER) of this simulation, the essential input to the PATHS procedure can be terminated. If no data are to be examined or modified, the PATHS procedure is executed using the user-specified input, the facility graph input, and the values that have been generated by default (DFLT); the next display will be that for the output summary.

---

*RANF[12] is the random number generator used in PATHS. Normally, an odd number, or better yet a prime, is used for the seed in order to maximize the period of the generated sequence.

This is not the case in the example of Figure 8. An examination or modification is to be made, and the first modification involves a node detection probability (type code = 2). The subject node is the target node labeled 1. In response to the request for a distribution code for the node 1 detection probability, the triangular distribution (3) has been specified. This distribution code is ignored (only 0 or 1 is acceptable) because, of course, the distribution must be deterministic. The procedure responds with a request for the (new) deterministic value (which it will do for any code except 0). The value is modified from 0 to 1 by the user. Although unnecessary, the modification is next examined by again entering type code 2 followed by label 1. Again the procedure responds with a deterministic value that is the floating point equivalent of the integer value that was previously specified. A distribution code of 0 is then entered to indicate that no further changes are to be made to the distribution or the value for node 1.

Type code 3 shows that an arc traversal time is a candidate for modification or examination. The user enters the node label pair 1,20 (or 20,1; the order is immaterial). If there were no arc joining nodes 1 and 20, the procedure would respond:

```
(1,20) IS NOT AN ARC
ENTER:  NODE LABEL PAIR
```

If the arc exists, the response is as shown in Figure 8. Normally the specification of a node label pair (such as 1,20) results in an unsurprising response from PATHS. There is either an arc joining the pair or there is not. Sometimes, however, specification of a node pair which is known to have a joining arc is rejected. For example, given a path with the nodal sequence (1,20,14,8), the arc associated with the pair (14,8) must exist but specification of the node pair 14,8 will cause PATHS to respond with

```
(14,8) IS NOT AN ARC
ENTER:  NODE LABEL PAIR
```

This unexpected response occurs because node 14 is a pseudo-node (a node used to break into two segments what otherwise would be a single straight path segment though an obstacle). The node pair must be specified by 1014,8; that is, 1000 must be added to the label of the pseudo-node.*

In the example under consideration, the arc joining the target node 1 to the barrier node 20 exists, and the procedure displays the current values (default values) of the triangular distribution that is assigned to the arc traversal time. Note that min = 1/2 mode and max = 3/2 mode; this relationship was previously fixed by default (DFLT). If examination of these values were sufficient, the next distribution code to be entered would be 0, but here the user wishes to retain the assigned distribution and alter only the values of the quantities that characterize it. Therefore 3 is entered followed by the values for mode, min, and max. Difficulty occurs because the magnitudes of the values entered do not have the proper relationship, as evidenced by the PATHS response. Entry of 2,1,3 is acceptable, but 3,1,5 would also have been acceptable. This makes evident that, although min must be less than mode and mode must be less than max, the exact proportions, 2:1:3, need not be retained; that is, the distribution need not be symmetrical about the mode.

The new values are examined by re-entering the type code 3 and node label pair 1,20; PATHS echoes the new values. Because nothing more is to be done with this arc, a 0 is entered for the response to the distribution code request. (The user may want to change the detection probability for this arc; PATHS will not recompute the arc detection probability using the mode of the new arc traversal time). The next user response to the type code request is 0 and therefore all modifications or examinations are terminated. The title for this simulation, I EXAMPLE, is entered when PATHS requests a header, indicating that user input is complete. The complete PATHS procedure is then executed, and, until all computations are completed, no interaction with PATHS is possible. When the output is ready for display, this fact will be signaled with CLEAR SCREEN AND PRESS RETURN TO CONTINUE. Output will begin with the "PATHS Summary."

---

*What is even more disconcerting is that 20,14 is acceptable, yet 20,1014 is not. It is therefore suggested that the user work with known arcs (evidenced by previous simulation(s) and if a known node pair is rejected, try the other three combinations (with 1000 added).

Figure 9 shows the input for essentially the same simulation that was carried out for Figure 8. In this example, however, no modifications are made and the brief input mode is used. No significant differences can be seen until PATHS requests the seed, the number of Phase 1 and Phase 2 replications, and the distribution codes for the node traversal, arc traversal, and guard response times. These are entered by the user in one input operation. It must be pointed out that entering the null distribution code, 0, is an inappropriate input in this context. The null distribution code is used only during the modification and examination portion of the procedure. If null distributions are entered, the procedure will abort, and the system will respond with a notification of a FATAL ERROR followed by READY. The user response to READY is to start the simulation again by entering CALL, UPATH.

Figure 10 shows the extended mode input when time is the measure. Again, target 1 is under consideration. The input is similar to the example in Figure 8 for the interruption measure except that the number of replications per path saved in the evaluation phase (Phase 2) is now fixed internally by the procedure and need not be specified. In addition, the response time distribution is fixed (deterministic) and the value is infinite (BIG); no distribution code need or can be specified for the response time. This fact is emphasized in the attempt to modify the response time distribution or its value. The user specification of 5 in the modification procedure is rejected, and the procedure responds with CANNOT MODIFY RESPONSE TIMES. The value for the target node detection probability is, however, altered as was done in Figure 8.* After an attempt to change the detection probability distribution from deterministic to lognormal (which is rejected), the value is changed from 0 to 1 and then checked. The traversal time for the target node is also examined but it is not modified (distribution code = 0) and modifications are halted (type code = 0). User-specified input is complete with the entry of the title T EXAMPLE.

The traversal time simulation is repeated in Figure 11 using the brief mode. No modifications are made and the seed, number of pathfinding (Phase 1) replications, and distributions for the node and arc traversal times are entered in one input operation. Following the specification of the title, the NOS response, JOB ACTIVE, appears. This is a response to a user request for the status of the job and is brought about by pressing "RETURN" during the computations; this action represents an interaction with the NOS rather than the PATHS procedure.

---

*This alteration is merely an exercise. Because the measure is time, the values for the detection probabilities are not material to the simulation.

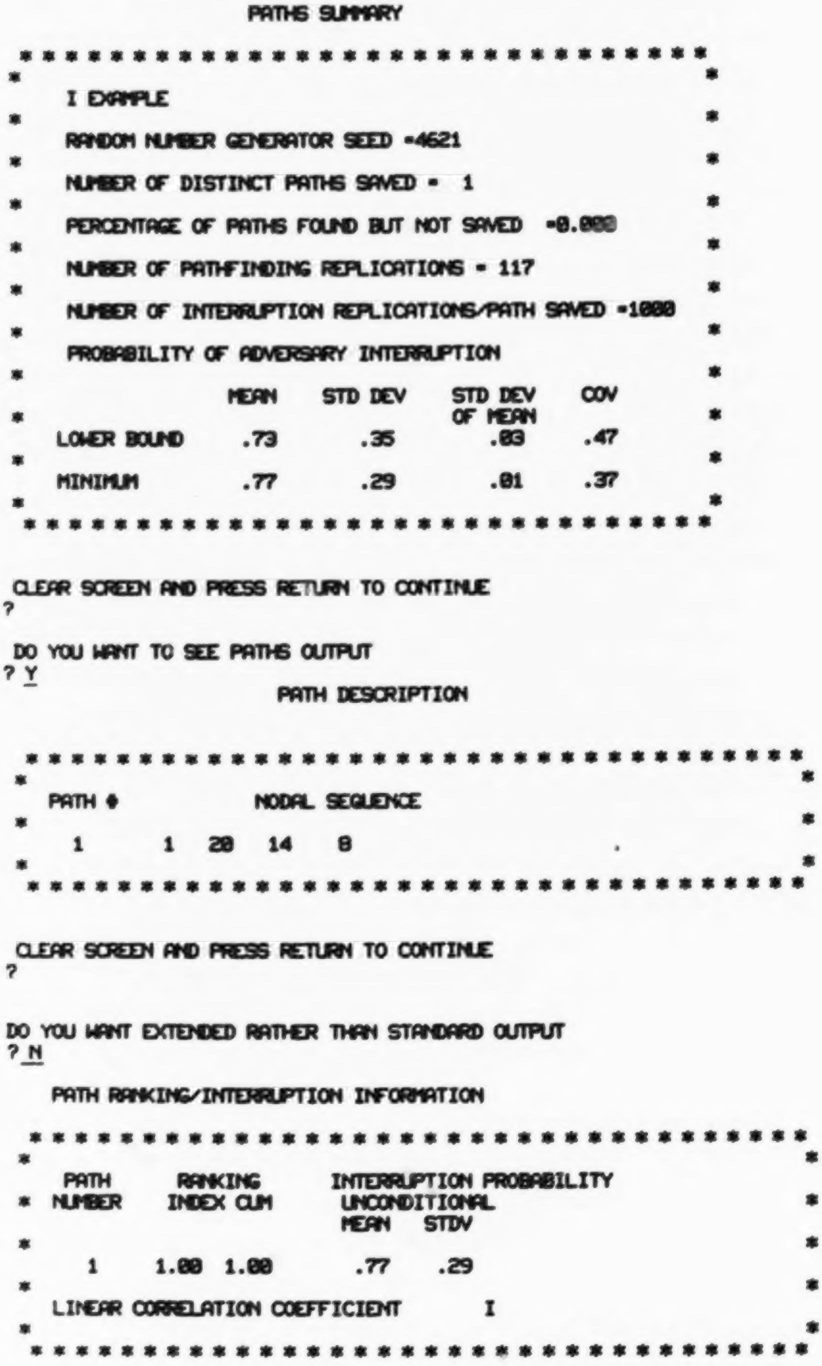
Should the execution of PATHS exceed the time limit (greater than ~63 seconds of central processor time), the system will display *TIME LIMIT*. The user's response is T,100. This command will allow the execution to continue for an extended time interval. (The need for additional computational time, when it arises, most often occurs when the number of Phase 1 pathfinding replication is large and mutiple targets are specified.)

The PATHS procedure can be halted at any point by pressing the "BREAK" key on the terminal or by typing STOP in response to the PATHS prompt (?). If "BREAK" is used, the system responds with *INTERRUPTED*. The user then types STOP and the system displays *TERMINATED* followed by READY on a new line. READY is a prompt in one of the NOS modes; it indicates that the system is awaiting the next command (for example: CALL,UPATH).

Output of PATHS

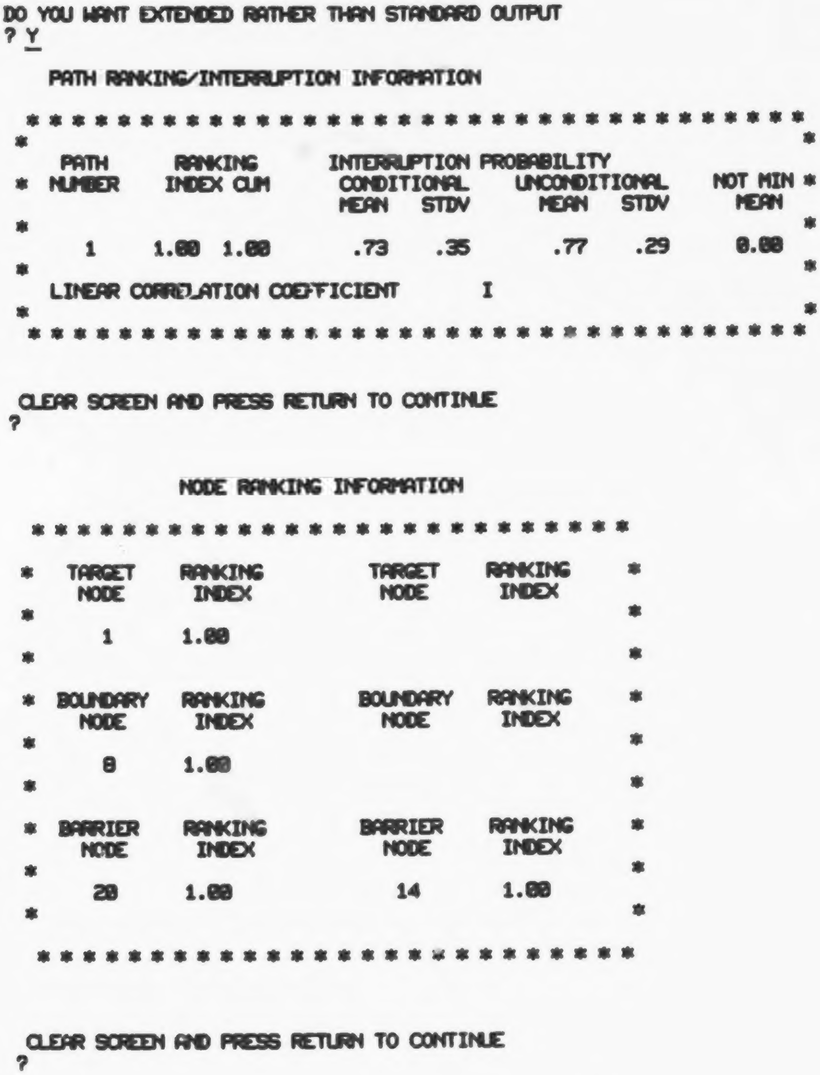
The outputs of PATHS for the simulations in which no modifications have been made, i.e., for the inputs shown in Figure 9 (interruption) and Figure 10 (time), are shown in Figures 12 and 13. Note that the examples in these figures demonstrate few instances of interaction between the user and the PATHS procedure. During the output sequence only two significant user responses are called for, i.e., those electing the amount (more than a summary) and the content (standard or extended) of the output. At the completion of the output (end of "Node Ranking Information"), the PATHS procedure displays a CLEAR SCREEN AND PRESS RETURN TO CONTINUE and then returns the user to a NOS mode (READY prompt).

PATHS Summary -- The "PATHS Summaries" of Figures 12 and 13 echo the user inputs for SEED, MRUN, and IRUN. In Figure 12, one path is saved, and in Figure 13, seven paths are saved. In general, the number of paths generated equals the number saved plus the number dumped. Because these outputs indicate that no paths are dumped, subsequent output



```
                    PATHS SUMMARY

 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 *                                                          *
    I EXAMPLE
 *                                                          *
    RANDOM NUMBER GENERATOR SEED =4621
 *                                                          *
    NUMBER OF DISTINCT PATHS SAVED =  1
 *                                                          *
    PERCENTAGE OF PATHS FOUND BUT NOT SAVED  =0.000
 *                                                          *
    NUMBER OF PATHFINDING REPLICATIONS = 117
 *                                                          *
    NUMBER OF INTERRUPTION REPLICATIONS/PATH SAVED =1000
 *                                                          *
    PROBABILITY OF ADVERSARY INTERRUPTION
 *                                                          *
                    MEAN     STD DEV    STD DEV     COV
 *                                     OF MEAN              *
   LOWER BOUND      .73        .35        .03      .47
 *                                                          *
   MINIMUM          .77        .29        .01      .37
 *                                                          *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?

 DO YOU WANT TO SEE PATHS OUTPUT
? Y
                      PATH DESCRIPTION

 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 *                                                                *
   PATH #            NODAL SEQUENCE
 *                                                                *
     1       1  20  14   8
 *                                                                *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?

DO YOU WANT EXTENDED RATHER THAN STANDARD OUTPUT
? N

    PATH RANKING/INTERRUPTION INFORMATION

 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 *                                                                *
    PATH      RANKING      INTERRUPTION PROBABILITY
 *  NUMBER    INDEX CUM      UNCONDITIONAL                        *
                             MEAN   STDV
 *                                                                *
      1     1.00 1.00        .77    .29
 *                                                                *
    LINEAR CORRELATION COEFFICIENT        I
 *                                                                *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

Figure 12. PATHS Output for an Interruption Simulation Showing Both Standard and Extended Content



```
DO YOU WANT EXTENDED RATHER THAN STANDARD OUTPUT
? Y

   PATH RANKING/INTERRUPTION INFORMATION

 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 *                                                                     *
    PATH      RANKING       INTERRUPTION PROBABILITY
 *  NUMBER    INDEX CUM      CONDITIONAL     UNCONDITIONAL    NOT MIN  *
                             MEAN   STDV      MEAN   STDV      MEAN
 *                                                                     *
      1     1.00  1.00        .73    .35       .77    .29      0.00
 *                                                                     *
    LINEAR CORRELATION COEFFICIENT        I
 *                                                                     *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?

               NODE RANKING INFORMATION

  * * * * * * * * * * * * * * * * * * * * * * * * * * *

  *   TARGET    RANKING         TARGET    RANKING      *
       NODE      INDEX           NODE      INDEX
  *                                                    *
        1       1.00
  *                                                    *

  *  BOUNDARY   RANKING        BOUNDARY   RANKING      *
       NODE      INDEX           NODE      INDEX
  *                                                    *
        8       1.00
  *                                                    *

  *  BARRIER    RANKING        BARRIER    RANKING      *
       NODE      INDEX           NODE      INDEX
  *                                                    *
       20       1.00              14      1.00
  *                                                    *

  * * * * * * * * * * * * * * * * * * * * * * * * * * *

 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?
```

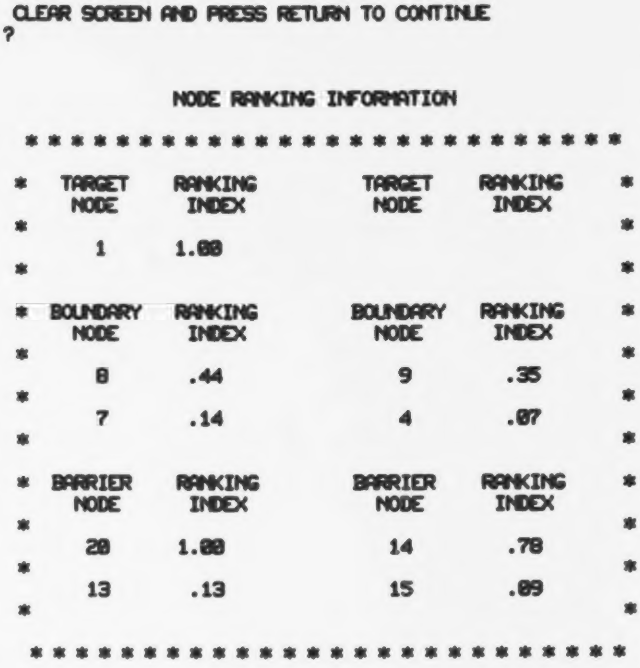
Figure 12. PATHS Output for an Interruption Simulation Showing Both Standard and Extended Content (Concluded)

```
                    PATHS SUMMARY

 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*                                                            *
    T EXAMPLE
*                                                            *
    RANDOM NUMBER GENERATOR SEED =4621
*                                                            *
    NUMBER OF DISTINCT PATHS SAVED =  7
*                                                            *
    PERCENTAGE OF PATHS FOUND BUT NOT SAVED  =0.000
*                                                            *
    NUMBER OF PATHFINDING REPLICATIONS = 117
*                                                            *
   UNCONDITIONAL MEAN PATH TRAVERSAL TIME
*                                                            *
         MINIMUM =   4.51
*                                                            *
           STDV  =    .49
*                                                            *
    (RATIO OF LOWER TO UPPER BOUND =  .95 )
*                                                            *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?

DO YOU WANT TO SEE PATHS OUTPUT
? Y
                      PATH DESCRIPTION


 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*                                                                    *
   PATH #             NODAL SEQUENCE
*                                                                    *
     1        1   20   14    8
*                                                                    *
     2        1   20   14    9
*                                                                    *
     3        1   20   15    7
*                                                                    *
     4        1   20   13    9
*                                                                    *
     5        1   20   14    7
*                                                                    *
     6        1   20   13    4
*                                                                    *
     7        1   20   14    4
*                                                                    *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?
```

Figure 13. PATHS Output for a Time Simulation Showing Both Standard and Extended Output

```
DO YOU WANT EXTENDED RATHER THAN STANDARD OUTPUT
? N

              PATH RANKING/TIME INFORMATION

 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*                                                                   *
 PATH   RANKING           PATH TRAVERSAL TIMES
* NMBR    INDEX CUM         UNCONDITIONAL                           *
                              MEAN   STDV
*                                                                   *
     1     .44  .44           4.51    .49
*                                                                   *
     2     .26  .70           4.64    .50
*                                                                   *
     3     .09  .79           4.81    .49
*                                                                   *
     4     .09  .89           4.70    .49
*                                                                   *
     5     .04  .93           4.91    .53
*                                                                   *
     6     .03  .97           4.99    .51
*                                                                   *
     7     .03 1.00           5.12    .57
*                                                                   *
   LINEAR CORRELATION COEFFICIENT     .90
*                                                                   *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?

DO YOU WANT EXTENDED RATHER THAN STANDARD OUTPUT
? Y

              PATH RANKING/TIME INFORMATION

 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*                                                                   *
 PATH   RANKING           PATH TRAVERSAL TIMES
* NMBR    INDEX CUM         CONDITIONAL       UNCONDITIONAL    NOT MIN *
                              MEAN   STDV      MEAN   STDV      MEAN
*                                                                   *
     1     .44  .44           4.23    .49      4.51    .49      4.74
*                                                                   *
     2     .26  .70           4.20    .42      4.64    .50      4.77
*                                                                   *
     3     .09  .79           4.48    .56      4.81    .49      4.85
*                                                                   *
     4     .09  .89           4.34    .36      4.70    .49      4.82
*                                                                   *
     5     .04  .93           4.47    .42      4.91    .53      4.93
*                                                                   *
     6     .03  .97           4.46    .47      4.99    .51      5.01
*                                                                   *
     7     .03 1.00           4.12    .32      5.12    .57      5.16
*                                                                   *
   LINEAR CORRELATION COEFFICIENT     .90
*                                                                   *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

Figure 13. PATHS Output for a Time Simulation Showing Both Standard and Extended Output (Continued)



```
CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?

               NODE RANKING INFORMATION

 * * * * * * * * * * * * * * * * * * * * * * * * * *

 *   TARGET    RANKING          TARGET    RANKING    *
      NODE      INDEX            NODE      INDEX
 *                                                   *
       1       1.00
 *                                                   *

 *  BOUNDARY   RANKING         BOUNDARY   RANKING    *
      NODE      INDEX            NODE      INDEX
 *                                                   *
       8        .44               9         .35
 *                                                   *
       7        .14               4         .07
 *                                                   *

 *  BARRIER    RANKING         BARRIER    RANKING    *
      NODE      INDEX            NODE      INDEX
 *                                                   *
      20       1.00              14         .78
 *                                                   *
      13        .13              15         .09
 *                                                   *

 * * * * * * * * * * * * * * * * * * * * * * * * * *
```

Figure 13. PATHS Output for a Time Simulation Showing Both Standard and Extended Output (Concluded)

under "Path Description" and "Path Ranking," which normally apply only to saved paths, will apply to all Phase 1 generated paths in these two examples.

For the summary interruption output (Figure 12), the facility estimates, i.e., the lower and upper bounds (the means for LOWER BOUND and MINIMUM), are listed along with the estimates of precision under PROBABILITY OF ADVERSARY INTERRUPTION. For SEED = 4621, the value of the lower bound is less than the upper bound* (LB < MIN), and therefore the Phase 1 sample size (MRUN = 117) appears to be adequate although the standard error or standard deviation of the lower bound is three times the standard error of the upper bound. For the traversal time output (Figure 13), the ratio of the lower to upper bound is less than 1, and therefore the Phase 1 sample size appears to be adequate in this example also.

Path Description -- At the end of the summary output, additional output will be displayed at the option of the user. In the examples in Figures 12 and 13, this output is requested. The "Path Descriptions" that follow provide the nodal sequence from target to boundary for each path saved (or, in these circumstances, generated). For the specified inputs, the path from target node 1 to boundary node 8 through barrier nodes 20 and 14 is both the minimum interruption probability and minimum traversal time path.

Path Ranking/Interruption or Time Information -- "Path Ranking/Interruption or Time Information" can be obtained with either extended or standard content by entering Y or N in response to the query: DO YOU WANT EXTENDED RATHER THAN STANDARD OUTPUT? The position of the standard or the extended output is between "Path Description" and "Node Ranking Information" in the output sequence. In the examples of Figures 12 and 13, both the standard and the extended output are shown in this position although only the one output requested appears during normal operation. Because the extended output includes the information contained in the standard output plus added conditional information, the following discussion addresses the extended output.

The extended "Path Ranking/Interruption Information" of Figure 12 lists the estimates for the mean conditional (given that the path is the minimum path) and mean (unconditional) probabilities of interruption. Because only one path is generated, these values are identical to the values for the lower and upper bounds that are the facility estimates in the summary output. It is of course expected that the upper bound value will be identical to the Phase 2 criticality estimate for some path, but the fact that the lower bound value equals the conditional value and is indeed representative of a real path obtains because only a single path is generated. Moreover, with a single generated path, the lower bound would be expected to equal the upper bound if the Phase 1 and Phase 2 samples had been statistically identical. The derived estimate of mean conditional (given that the path is not the minimum path) interruption probability (NOT MIN) is set equal to zero despite the observed difference in values for the bounds because the path rank (RANKING INDEX) is indeed 1. This ranking index also indicates why the linear correlation coefficient is indeterminate (I). There are no other paths with which to attempt a correlation.

---

*For a different seed, this may not be the case.

For the extended "Path Ranking/Time Information" of Figure 13, the Phase 2 path criticality estimate (estimate of the unconditional mean) for path 1 equals the facility estimate, MINIMUM, and this physical path is the least traversal time path to target 1 for the specified input. The estimate for path 1 (4.51 minutes) is less than those that result for all other (less critical) paths. The linear correlation between the Phase 2 path criticality estimates and the Phase 1 path criticality estimates (RANKING INDEX) for all saved paths is excellent. The coefficient is not unity because path 3, although somewhat more likely to be generated than path 4 (accounting for round-off errors), has an (unconditional) mean traversal time which is greater than that of path 4. In other words, path length in relation to path rank appears to be anomalous for this pair of paths, and this situation is reflected in the value of the linear correlation coefficient.

For both the interruption and time outputs, the cumulative ranking (CUM) that is displayed with the information for the last path listed is 1. This total is less than 1 if any paths are dumped. When a dump occurs a "path number," D, is displayed under the last real path listed, and the D is followed by the dump rank, the CUM for all paths both saved and dumped, and the estimates (mean and standard deviation) for the conditional interruption probability (or traversal time) mass that is in the dump. No unconditional information is displayed because the dump is not evaluated in Phase 2.

Node Ranking Information -- The "Node Ranking Information" of Figure 12 recapitulates what is already obvious, namely, that only the nodes of path 1 are generated and that the rank of every node is equal to the path rank. If the dump were not empty, additional nodes would be displayed and ranked. To demonstrate what might occur, assume for the example of Figure 13 that only path 1 is saved and all other paths (2 through 7) are dumped. The "Node Ranking Information" would be identical to that in Figure 13. The "Path Ranking/Time Information", in addition to the path 1 information would include

```
D   .56   1.00   4.18   .42
```

The nodes that appear in the "Node Ranking Information", as a result of the dump, are nodes 4, 7, 9, 13, and 15. Because of the artifice being used, the dumped paths are visible, and it is apparent that the rank of node 4 is equal to the sum of the ranks of paths 6 and 7 in which node 4 occurs. Similarly, the rank of node 7 is equal to the sum of the ranks of paths 3 and 5. This pattern is demonstrated for all the nodes generated.

Examples and Discussion

Figures 14 and 15 show outputs for interruption and time simulations for which, compared to previous examples, only the target node (label 201) is different. Unless it is otherwise noted, in these examples as well as the other examples cited in this section, all response times will equal 2 minutes (for interruption simulations) and all distribution codes will be designated 3 (prior probability distributions are triangular).

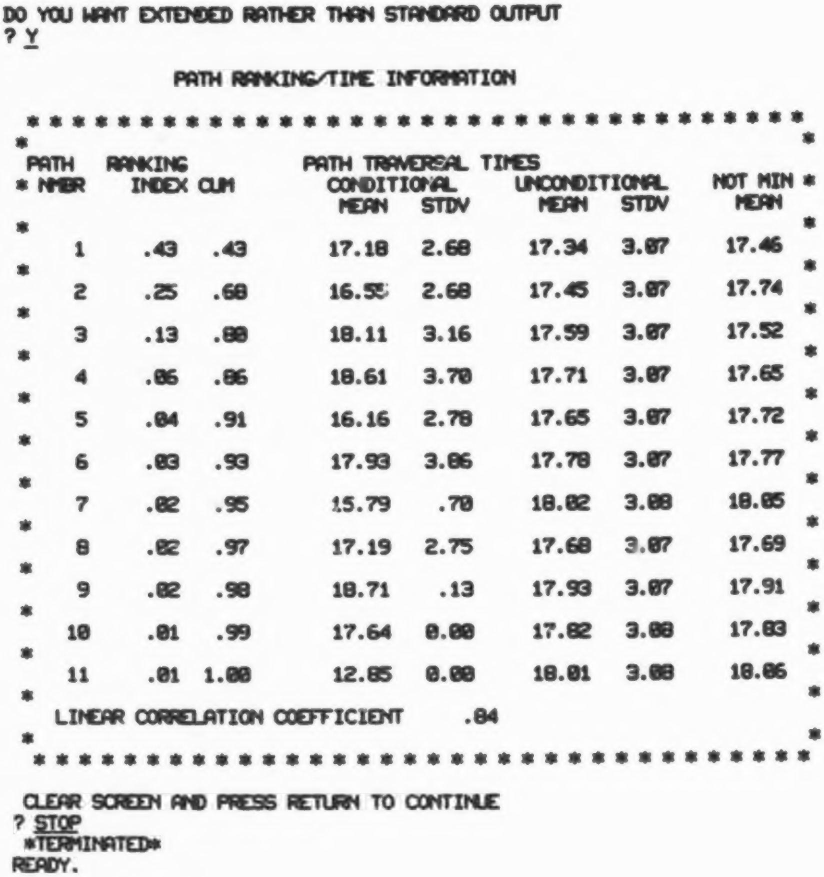
Interval, Precision, and the Security Protection System -- In Figure 14, the "PATHS Summary" illustrates by the lower and the upper bounds, [0.98, 0.98], what was pointed out in Section 2, page 30; namely, that the interval for the bounds and the values for the associated estimates of precision may be expected to be small where the expected value for the facility minimum is close to 1 (or 0). Therefore, any modification of the response times or detection probabilities that increases the security of the facility will increase precision and decrease the interval of indeterminacy defined by the bounds.

```
                       PATHS SUMMARY

  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 *                                                             *
     FIGURE 14 (TARGET 201)
 *                                                             *
     RANDOM NUMBER GENERATOR SEED =4621
 *                                                             *
     NUMBER OF DISTINCT PATHS SAVED =  1
 *                                                             *
     PERCENTAGE OF PATHS FOUND BUT NOT SAVED   =0.000
 *                                                             *
     NUMBER OF PATHFINDING REPLICATIONS = 117
 *                                                             *
     NUMBER OF INTERRUPTION REPLICATIONS/PATH SAVED =1000
 *                                                             *
     PROBABILITY OF ADVERSARY INTERRUPTION
 *                                                             *
                    MEAN     STD DEV     STD DEV     COV
 *                                      OF MEAN                *
    LOWER BOUND      .98         .00         .00     .00
 *                                                             *
    MINIMUM          .98         .00         .00     .00
 *                                                             *
  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?

 DO YOU WANT TO SEE PATHS OUTPUT
? Y
                        PATH DESCRIPTION

  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 *                                                                     *
    PATH #             NODAL. SEQUENCE
 *                                                                     *
      1      201  213  100  107  104   10    6
 *                                                                     *
  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?

DO YOU WANT EXTENDED RATHER THAN STANDARD OUTPUT
? Y

    PATH RANKING/INTERRUPTION INFORMATION

  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 *                                                                     *
    PATH       RANKING       INTERRUPTION PROBABILITY
 *  NUMBER     INDEX CUM       CONDITIONAL     UNCONDITIONAL    NOT MIN *
                               MEAN   STDV      MEAN   STDV      MEAN
 *                                                                     *
      1      1.00 1.00         .98    .00       .98    .00       0.00
 *                                                                     *
    LINEAR CORRELATION COEFFICIENT        I
 *                                                                     *
  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

Figure 14. PATHS Output for a Single Target Interruption Simulation

```
                 PATHS SUMMARY
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*                                                             *
    FIGURE 15 (TARGET 201)
*                                                             *
    RANDOM NUMBER GENERATOR SEED =4621
*                                                             *
    NUMBER OF DISTINCT PATHS SAVED = 11
*                                                             *
    PERCENTAGE OF PATHS FOUND BUT NOT SAVED  =0.000
*                                                             *
    NUMBER OF PATHFINDING REPLICATIONS = 117
*                                                             *
   UNCONDITIONAL MEAN PATH TRAVERSAL TIME
*                                                             *
        MINIMUM =  17.34
*                                                             *
          STDV  =   3.07
*                                                             *
    (RATIO OF LOWER TO UPPER BOUND =  .99 )
*                                                             *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?

DO YOU WANT TO SEE PATHS OUTPUT
? Y
                    PATH DESCRIPTION

 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*                                                                   *
  PATH #          NODAL SEQUENCE
*                                                                   *
    1     201  206   10    6
*                                                                   *
    2     201  205   10    6
*                                                                   *
    3     201  207   11    4
*                                                                   *
    4     201  216  218  210   13    9
*                                                                   *
    5     201  207   12    4
*                                                                   *
    6     201  207   11    5
*                                                                   *
    7     201  206   10    5
*                                                                   *
    8     201  207   10    6
*                                                                   *
    9     201  216  218  210   13    4
*                                                                   *
   10     201  207   11    6
*                                                                   *
   11     201  207   11    9
*                                                                   *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?
```

Figure 15. PATHS Output for a Single Target Time Simulation

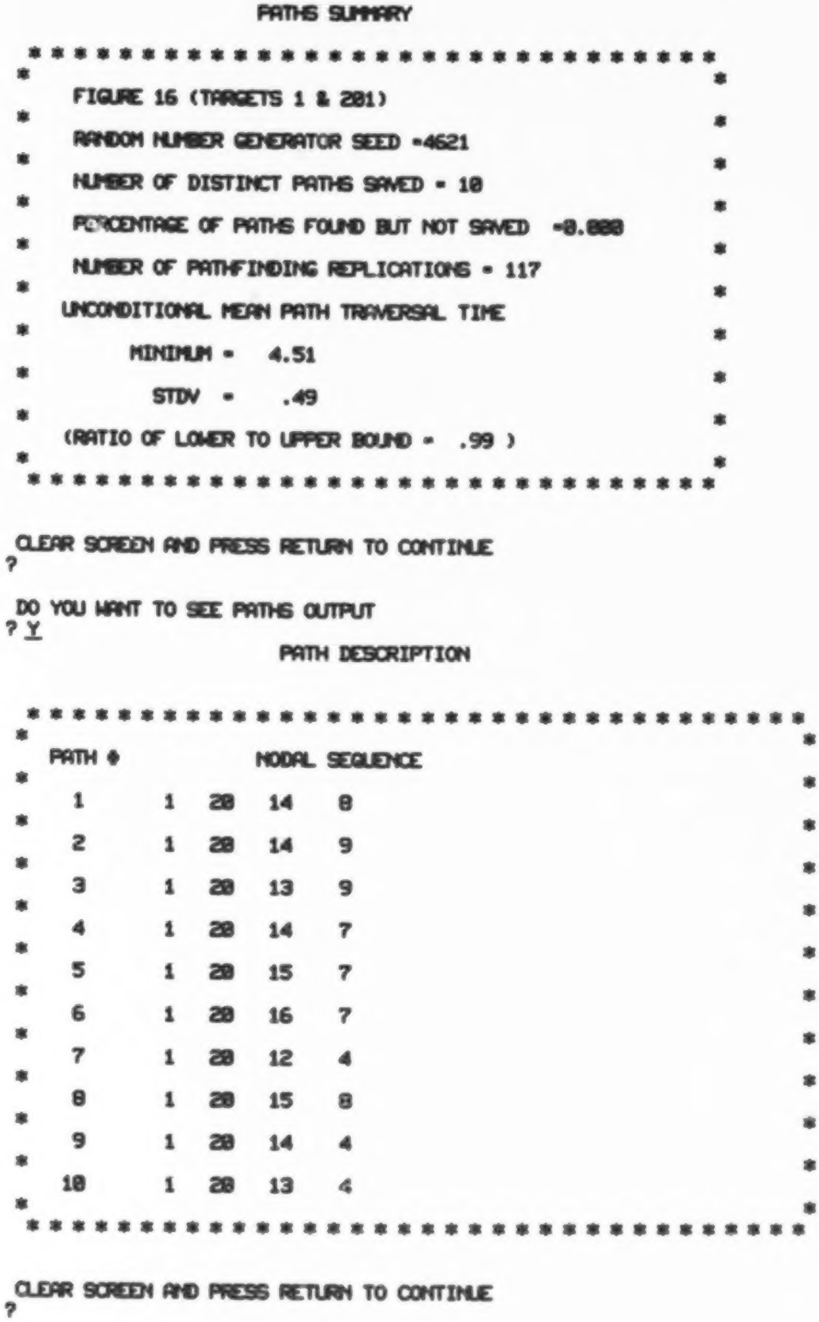


```
DO YOU WANT EXTENDED RATHER THAN STANDARD OUTPUT
? Y

             PATH RANKING/TIME INFORMATION

 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*                                                                  *
 PATH   RANKING            PATH TRAVERSAL TIMES
* NMBR    INDEX CUM          CONDITIONAL      UNCONDITIONAL     NOT MIN *
                               MEAN   STDV       MEAN   STDV       MEAN
*                                                                  *
     1      .43   .43        17.10   2.60       17.34   3.07       17.46
*                                                                  *
     2      .25   .68        16.55   2.60       17.45   3.07       17.74
*                                                                  *
     3      .13   .80        18.11   3.16       17.59   3.07       17.52
*                                                                  *
     4      .06   .86        18.61   3.70       17.71   3.07       17.65
*                                                                  *
     5      .04   .91        16.16   2.70       17.65   3.07       17.72
*                                                                  *
     6      .03   .93        17.93   3.06       17.78   3.07       17.77
*                                                                  *
     7      .02   .95        15.79    .70       18.02   3.08       18.05
*                                                                  *
     8      .02   .97        17.19   2.75       17.68   3.07       17.69
*                                                                  *
     9      .02   .98        18.71    .13       17.93   3.07       17.91
*                                                                  *
    10      .01   .99        17.64   0.00       17.82   3.08       17.83
*                                                                  *
    11      .01  1.00        12.85   0.00       18.01   3.08       18.06
*                                                                  *
   LINEAR CORRELATION COEFFICIENT     .04
*                                                                  *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
? STOP
 *TERMINATED*
READY.
```

Figure 15. PATHS Output for a Single Target Time Simulation (Concluded)

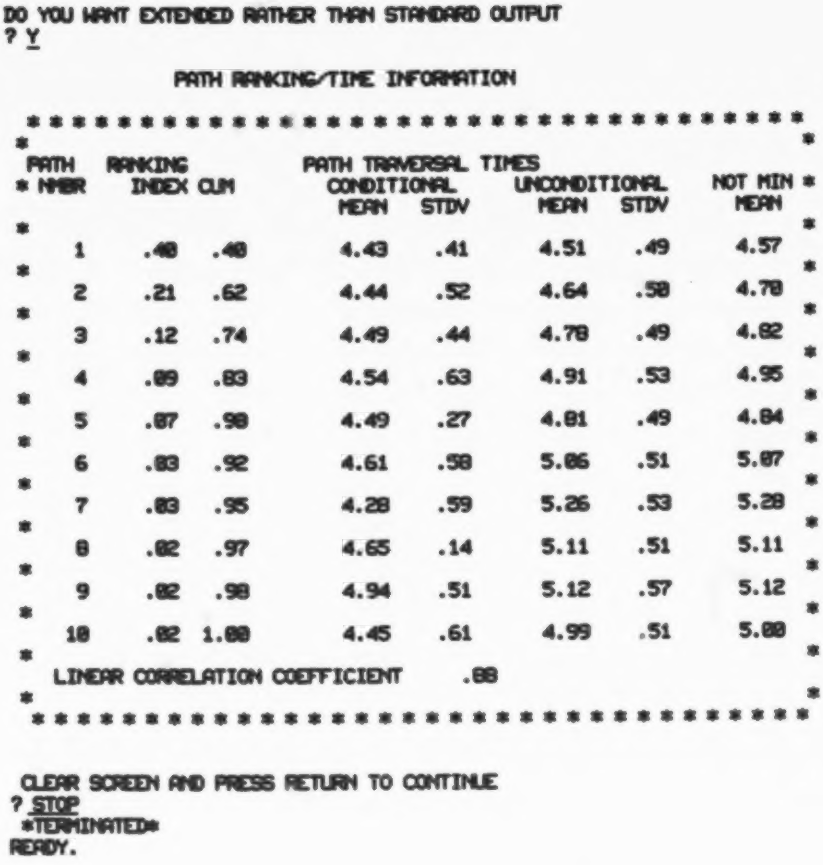
Normal and Extended Content in the Path Ranking Information -- In Figure 15, the extended "Path Ranking/Time Information", in particular that for path 11 (see also path 7 of Figure 13), shows that the path least likely to be generated (RANKING INDEX = 0.01) is the least time path when it is generated (CONDITIONAL MEAN = 12.85 minutes). Nonetheless, because the estimate of the unconditional mean is large (compared to estimates of all other unconditional means), path 11 is, on the average, not significant to facility performance under this measure. Indeed, conditional estimates (CONDITIONAL MEAN and MEAN of NOT MIN), though instructive, are not often necessary to the understanding of the output. Normally, path ranking information with standard content is sufficent for providing adequate information about a simulation, particularly when the linear correlation is good, as in Figure 15.

Comparison of Multiple and Single Target Simulations -- In the examples in Figures 14 and 15, the minimum traversal time and minimum interruption probability paths are not the same (unlike the examples of Figures 12 and 13), and this fact is noted to emphasize the dependence of facility performance on measure. More interesting is a comparison of the most critical path of Figures 12 and 13 with those of Figures 14 and 15. On the basis of the estimates obtained when the targets are considered singly, it might be expected that, for time and interruption simulations in which paths to both targets (1 and 201) are examined in competition, the path (1,20,14,8) would be the most critical with respect to both time and interruption. The outputs for the two-target (1 and 201) simulations of Figures 16 and 17 confirm this expectation. For the time simulation (Figure 16), path (1,20,14,8) is not only the least (unconditional) traversal time path but also the most frequently generated one (it is at least 1.9 times as likely to be the minimum as any other path). For the interruption simulation (Figure 17), path 2 (1,20, 14,8) is the path having the least (unconditional) probability of interruption, but it is not the path that is most likely to be generated. Indeed, it is ranked second and this situation is indicated by the anticorrelation of the ranks and (unconditional) interruption probabilities of Figure 17. The path of higher rank, although more likely to be the minimum, has a greater probability (0.98 > 0.77) of being interrupted when the paths are regarded conditionally as minimum paths and about the same interruption probability (0.98 ~ 1.00) when the paths are regarded conditionally as not minimum paths; therefore, path 1 is likely to be less critical than path 2 with respect to interruption.



```
                    PATHS SUMMARY

 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*                                                             *
    FIGURE 16 (TARGETS 1 & 201)
*                                                             *
    RANDOM NUMBER GENERATOR SEED =4621
*                                                             *
    NUMBER OF DISTINCT PATHS SAVED = 10
*                                                             *
    PERCENTAGE OF PATHS FOUND BUT NOT SAVED  =0.000
*                                                             *
    NUMBER OF PATHFINDING REPLICATIONS = 117
*                                                             *
   UNCONDITIONAL MEAN PATH TRAVERSAL TIME
*                                                             *
         MINIMUM =   4.51
*                                                             *
           STDV =    .49
*                                                             *
   (RATIO OF LOWER TO UPPER BOUND =  .99 )
*                                                             *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?

 DO YOU WANT TO SEE PATHS OUTPUT
? Y
                        PATH DESCRIPTION

 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*                                                                   *
   PATH #            NODAL SEQUENCE
*                                                                   *
    1       1   20   14   8
*                                                                   *
    2       1   20   14   9
*                                                                   *
    3       1   20   13   9
*                                                                   *
    4       1   20   14   7
*                                                                   *
    5       1   20   15   7
*                                                                   *
    6       1   20   16   7
*                                                                   *
    7       1   20   12   4
*                                                                   *
    8       1   20   15   8
*                                                                   *
    9       1   20   14   4
*                                                                   *
    10      1   20   13   4
*                                                                   *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?
```

Figure 16. PATHS Output for a Two-Target Time Simulation



```
DO YOU WANT EXTENDED RATHER THAN STANDARD OUTPUT
? Y

                 PATH RANKING/TIME INFORMATION

 ***********************************************************
*                                                           *
 PATH   RANKING            PATH TRAVERSAL TIMES
* NMBR     INDEX CUM          CONDITIONAL      UNCONDITIONAL     NOT MIN *
                               MEAN   STDV       MEAN   STDV       MEAN
*                                                           *
    1     .40   .40           4.43    .41       4.51    .49       4.57
*                                                           *
    2     .21   .62           4.44    .52       4.64    .58       4.70
*                                                           *
    3     .12   .74           4.49    .44       4.78    .49       4.82
*                                                           *
    4     .09   .83           4.54    .63       4.91    .53       4.95
*                                                           *
    5     .07   .90           4.49    .27       4.81    .49       4.84
*                                                           *
    6     .03   .92           4.61    .58       5.06    .51       5.07
*                                                           *
    7     .03   .95           4.28    .59       5.26    .53       5.28
*                                                           *
    8     .02   .97           4.65    .14       5.11    .51       5.11
*                                                           *
    9     .02   .98           4.94    .51       5.12    .57       5.12
*                                                           *
   10     .02  1.00           4.45    .61       4.99    .51       5.00
*                                                           *
   LINEAR CORRELATION COEFFICIENT      .88
*                                                           *
 ***********************************************************

 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
? STOP
 *TERMINATED*
READY.
```

Figure 16. PATHS Output for a Two-Target Time Simulation (Concluded)

```
                    PATHS SUMMARY
 *******************************
*                                                *
   FIGURE 17 (TARGETS 1 & 201)
*                                                *
   RANDOM NUMBER GENERATOR SEED =4621
*                                                *
   NUMBER OF DISTINCT PATHS SAVED =  2
*                                                *
   PERCENTAGE OF PATHS FOUND BUT NOT SAVED  =0.000
*                                                *
   NUMBER OF PATHFINDING REPLICATIONS = 117
*                                                *
   NUMBER OF INTERRUPTION REPLICATIONS/PATH SAVED =1000
*                                                *
   PROBABILITY OF ADVERSARY INTERRUPTION
*                                                *
                  MEAN    STD DEV   STD DEV    COV
*                                   OF MEAN      *
  LOWER BOUND     .75       .30       .03      .40
*                                                *
  MINIMUM         .77       .30       .01      .39
*                                                *
 *******************************

CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?

DO YOU WANT TO SEE PATHS OUTPUT
? Y
                      PATH DESCRIPTION

 *********************************
*                                                      *
  PATH #          NODAL SEQUENCE
*                                                      *
   1     201  213  108  107  104   10    6
*                                                      *
   2       1   20   14    8
*                                                      *
 *********************************

CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?

DO YOU WANT EXTENDED RATHER THAN STANDARD OUTPUT
? Y

  PATH RANKING/INTERRUPTION INFORMATION

 *********************************
*                                                      *
   PATH     RANKING     INTERRUPTION PROBABILITY
*  NUMBER   INDEX CUM    CONDITIONAL   UNCONDITIONAL   NOT MIN *
                         MEAN  STDV     MEAN  STDV      MEAN
*                                                      *
    1       .61   .61    .98   .00      .98   .00       .98
*                                                      *
    2       .39  1.00    .41   .19      .77   .30      1.00
*                                                      *
  LINEAR CORRELATION COEFFICIENT   -1.00
*                                                      *
 *********************************
```

Figure 17. PATHS Output for a Two-Target Interruption Simulation

Output Resulting from Changes in MRUN and SEED -- Repeating the simulation of Figure 17 with an increase in the number of pathfinding replications (Figure 18) does not change the estimates for the means substantially. Nonetheless, increasing the Phase 1 sample size to 500 improves the precision estimates (the standard deviations of the Phase 1 and Phase 2 sample means are now both equal to 0.01), reduces the interval [LB, MIN] from [0.75, 0.77] to [0.77, 0.77], and alters slightly the ranking indices of the two generated paths--from 0.61 to 0.63 and from 0.39 to 0.37.

Another repetition of the example of Figure 17, this time with only the seed changed (Figure 19), results in the generation of a third path, but this simulation yields substantially the same estimates as those obtained in Figures 17 and 18. This is the case despite the fact that the lower exceeds the upper bound. Generation of path 3, however, attests to the fact that the Phase 1 sample size is sufficient to generate paths the likelihoods of which are at least 0.01 of being most critical. Thus, although the interval, [LB, MIN], and the precision estimates could be improved, this simulation supports the proposition that there is little chance of overlooking a critical path when the number (MRUN) of pathfinding replications is at least 117. When SEED = 827 and MRUN = 5000, three additional paths are generated (total = 6), but not one of these is more critical than path 3. The primary effect of increasing MRUN is an improvement in the precision estimates and an adjustment to the interval so that LB $\leq$ MIN. This adjustment alone can often be made by merely changing the seed (cf., 4621 of Figure 17).

Interpretation of Extended Path Ranking Information -- Perhaps the most interesting aspect of Figure 19 is that, if the conditions (the appropriate realization) causing the generation of path 3 are duplicated under actual physical conditions, then the conditional interruption probability along this path is very small (CONDITIONAL MEAN = 0.0). Even if path 3 is not the minimum path, the conditional interruption probability (MEAN of NOT MIN = 0.78) is relatively low; the corresponding estimate for path 2 is much higher (MEAN of NOT MIN = 0.94). Nonetheless, the average interruption probability for path 2 (UNCONDITIONAL

```
                        PATHS SUMMARY

 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*                                                              *
    FIGURE 18 (TARGETS 1 & 201)
*                                                              *
    RANDOM NUMBER GENERATOR SEED =4621
*                                                              *
    NUMBER OF DISTINCT PATHS SAVED =  2
*                                                              *
    PERCENTAGE OF PATHS FOUND BUT NOT SAVED  =0.000
*                                                              *
    NUMBER OF PATHFINDING REPLICATIONS = 500
*                                                              *
    NUMBER OF INTERRUPTION REPLICATIONS/PATH SAVED =1000
*                                                              *
    PROBABILITY OF ADVERSARY INTERRUPTION
*                                                              *
                     MEAN     STD DEV    STD DEV     COV
*                                        OF MEAN               *
   LOWER BOUND       .77        .29        .01       .37
*                                                              *
   MINIMUM           .77        .30        .01       .39
*                                                              *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?

 DO YOU WANT TO SEE PATHS OUTPUT
? Y
                          PATH DESCRIPTION

 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*                                                                      *
   PATH #            NODAL SEQUENCE
*                                                                      *
     1      201  213  108  107  104   10    6
*                                                                      *
     2        1   20   14    8
*                                                                      *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?

DO YOU WANT EXTENDED RATHER THAN STANDARD OUTPUT

? Y

   PATH RANKING/INTERRUPTION INFORMATION

 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*                                                                      *
   PATH      RANKING      INTERRUPTION PROBABILITY
*  NUMBER    INDEX CUM     CONDITIONAL     UNCONDITIONAL      NOT MIN  *
                           MEAN   STDV      MEAN   STDV        MEAN
*                                                                      *
     1       .63   .63      .98   .00        .98   .00          .98
*                                                                      *
     2       .37  1.00      .43   .18        .77   .30          .97
*                                                                      *
   LINEAR CORRELATION COEFFICIENT    -1.00
*                                                                      *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

Figure 18. PATHS Output for a Two-Target Interruption Simulation with Increased Pathfinding Replications

```
                    PATHS SUMMARY

 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*                                                             *
    FIGURE 19 (TARGETS 1 & 201)
*                                                             *
    RANDOM NUMBER GENERATOR SEED = 827
*                                                             *
    NUMBER OF DISTINCT PATHS SAVED =  3
*                                                             *
    PERCENTAGE OF PATHS FOUND BUT NOT SAVED  =0.000
*                                                             *
    NUMBER OF PATHFINDING REPLICATIONS = 117
*                                                             *
    NUMBER OF INTERRUPTION REPLICATIONS/PATH SAVED =1000
*                                                             *
    PROBABILITY OF ADVERSARY INTERRUPTION
*                                                             *
                  MEAN     STD DEV     STD DEV     COV
*                                      OF MEAN                *
   LOWER BOUND     .76       .29         .03       .39
*                                                             *
   MINIMUM         .75       .29         .01       .39
*                                                             *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?

DO YOU WANT TO SEE PATHS OUTPUT
? Y
                      PATH DESCRIPTION


 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*                                                                       *
  PATH #            NODAL SEQUENCE
*                                                                       *
    1     201  213  108  107  104   10    6
*                                                                       *
    2       1   20   14    8
*                                                                       *
    3       1   20   14    9
*                                                                       *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?
```

Figure 19. PATHS Output for a Two-Target Interruption Simulation with Changed Seed

```
DO YOU WANT EXTENDED RATHER THAN STANDARD OUTPUT
? Y

   PATH RANKING/INTERRUPTION INFORMATION

  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 *                                                                 *
    PATH      RANKING      INTERRUPTION PROBABILITY
 *  NUMBER    INDEX CUM     CONDITIONAL     UNCONDITIONAL    NOT MIN *
                            MEAN   STDV      MEAN   STDV      MEAN
 *                                                               1 *
      1        .61   .61     .98    .00      1.00    .00      1.00
 *                                                                 *
      2        .38   .99     .43    .17       .75    .29       .94
 *                                                                 *
      3        .01  1.00     .00   0.00       .77    .29       .78
 *                                                                 *
    LINEAR CORRELATION COEFFICIENT     -.73
 *                                                                 *
  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

1
 IF THIS DERIVED VALUE EXCEEDS 1.00, IT IS SET EQUAL TO 1.00.
```

Figure 19. PATHS Output for a Two-Target Interruption Simulation with Changed Seed (Concluded)

MEAN = MINIMUM = 0.75) is less than that for path 3, and therefore path 2 can be regarded as more critical than path 3.*

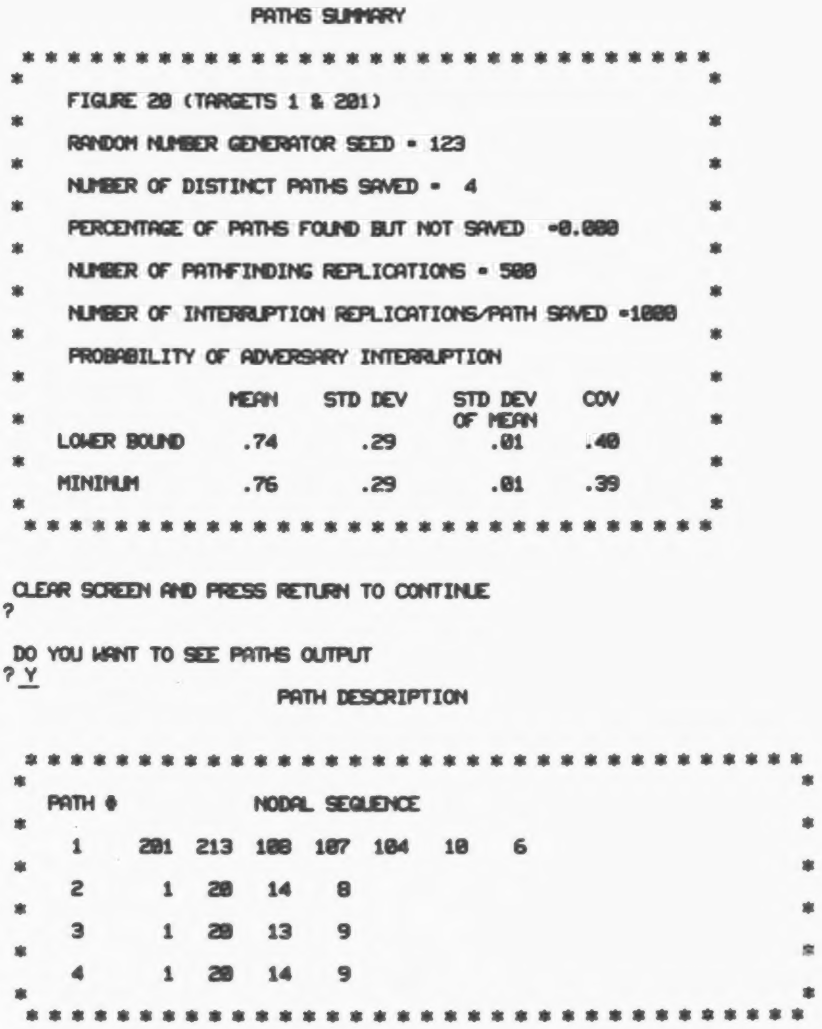
This contention for first place with respect to criticality extends to the example in Figure 20, although here the contending paths are labeled 2 and 4 rather than 2 and 3. Figure 20 shows the four paths that result when the seed is changed and MRUN is increased. Several aspects of this output are of interest. First, path 4 (formerly path 3) appears to be slightly more critical than path 2; path 4 has a lower conditional interruption probability (0.76) even though the conditions for which it is the minimum are infrequently realized. Even when path 4 is not a minimum, its conditional interruption probability (0.76) is less than the unconditional interruption probability for any other path. Indeed, this path, which is least likely to be the minimum, determines the facility upper bound (MINIMUM = 0.76). As a result, this simulation demonstrates that PATHS tends to provide a conservative upper bound.

To see why this is so, consider the situation in which path 4 is not generated. Path 3 is then indisputably most critical, the interval [LB, MIN] is [0.74, 0.78] and, given a facility threshold, P_T, of say 0.77, the facility is regarded as probably not secure. With the appearance of path 4, however, the facility is not secure; at least one path has been generated which violates the threshold.

Factors Influencing the Number of Paths Generated -- A second aspect of Figure 20 concerns the number of paths generated. In Figure 19, the generation of three paths (as opposed to two for Figures 17 and 18) could be attributed to the change of the seed. In Figure 20, four paths could result from a new seed (SEED = 123), an increase in MRUN, or both. In fact, it is the increase in Phase 1 sample size alone that is responsible. (A simulation identical to the example of Figure 20 except for MRUN = 117 results in two paths). In general, the number of paths generated depends not only on the seed and number of pathfinding replications but also upon the user-specified detection probabilities and

---

*It is necessary to mention that this "contest" between paths 2 and 3 is very much dependent upon the user's choice of prior probability distributions. If the user selects normal (4,4,4) rather than triangular (3,3,3) distributions for the example of Figure 19, then the output is essentially unchanged except for two estimates (UNCONDITIONAL MEAN = 0.99 and MEAN of NOT MIN = 1.00) for path 3. In this circumstance, all estimates support the fact that path 2 (1,20,14,8) is the most critical.



```
                         PATHS SUMMARY

  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 *                                                             *
     FIGURE 20 (TARGETS 1 & 201)
 *                                                             *
     RANDOM NUMBER GENERATOR SEED = 123
 *                                                             *
     NUMBER OF DISTINCT PATHS SAVED =  4
 *                                                             *
     PERCENTAGE OF PATHS FOUND BUT NOT SAVED  =0.000
 *                                                             *
     NUMBER OF PATHFINDING REPLICATIONS = 500
 *                                                             *
     NUMBER OF INTERRUPTION REPLICATIONS/PATH SAVED =1000
 *                                                             *
     PROBABILITY OF ADVERSARY INTERRUPTION
 *                                                             *
                    MEAN      STD DEV     STD DEV     COV
 *                                        OF MEAN              *
    LOWER BOUND      .74        .29         .01       .40
 *                                                             *
    MINIMUM          .76        .29         .01       .39
 *                                                             *
  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?
 DO YOU WANT TO SEE PATHS OUTPUT
? Y
                           PATH DESCRIPTION

  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 *                                                                   *
   PATH #               NODAL SEQUENCE
 *                                                                   *
     1     201  213  108  107  104   10    6
 *                                                                   *
     2       1   20   14    8
 *                                                                   *
     3       1   20   13    9
 *                                                                   *
     4       1   20   14    9
 *                                                                   *
  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

Figure 20. PATHS Output for a Two-Target Interruption Simulation with Changed Seed and Increased Pathfinding Replications

```
 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
?

DO YOU WANT EXTENDED RATHER THAN STANDARD OUTPUT
? Y

   PATH RANKING/INTERRUPTION INFORMATION

 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*                                                                    *
   PATH      RANKING      INTERRUPTION PROBABILITY
*  NUMBER    INDEX CUM      CONDITIONAL     UNCONDITIONAL     NOT MIN *
                            MEAN   STDV       MEAN   STDV      MEAN
*                                                                    *
     1       .56   .56       .98    .00        .98    .00        .98
*                                                                    *
     2       .43  1.00       .43    .17        .78    .20       1.00
*                                                                    *
     3       .00  1.00       .00   0.00        .99    .07        .99
*                                                                    *
     4       .00  1.00       .00   0.00        .76    .29        .76
*                                                                    *
   LINEAR CORRELATION COEFFICIENT     -.14
*                                                                    *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

 CLEAR SCREEN AND PRESS RETURN TO CONTINUE
? STOP
 *TERMINATED*
READY.
```

Figure 20. PATHS Output for a Two-Target Interruption Simulation with Changed Seed and Increased Pathfinding Replications (Concluded)

response times, number of targets examined, and prior probability distributions. Furthermore, it is not feasible to predict in advance what the number will be.

Analysis of Variance Resulting From Changes in MRUN and SEED -- A third and final aspect of Figure 20 concerns the significance of the estimates that are obtained. Disregarding the effect of sample size on the estimates of precision, all estimates in the simulations of Figures 17 through 20 appear to be remarkably consistent, with the possible exception of the ranking indices displayed in Figure 20. Although relative rank is preserved in all simulations, the frequency with which path 2 is generated in Figure 20 appears to be significantly increased, evidently at the expense of path 1. Whether this difference observed in Figure 20, or for that matter, whether any differences observed in all estimates (except precision estimates) for the simulations of Figures 17 through 20 are significant can be determined using standard analysis of variance (ANOVA) techniques (see Reference 13).

To this end, two-factor (SEED and MRUN), 3 x 3 factorial "experiments" were carried out without replication.* Observables are values (for SEED = 123, 827, 4621 and MRUN = 117, 500, 1000) of the ranking index of the path (1,20,14,8), the lower bound, and the upper bound that are obtained in nine two-target (1,201) simulations. Figure 21 reproduces the ANOVA tables for the three "experiments" that are used to test the hypothesis that the values for each observable are not significantly different. At the 0.10 level of significance, $F_{0.90}$ (2,4) = 4.32 which is greater than any of the F-values listed. Thus, the hypothesis of no difference in values for the observables cannot be rejected at this level or the 0.05 and 0.01 levels. At the 0.25 level, $F_{0.75}$ (2,4) = 2.00, and the hypothesis that the upper bound is independent of the seed (MRUN shouldn't and doesn't matter) can be rejected, but the chance of a Type I error is 1 in 4.

On the basis of these "experiments" then, it seems reasonable to conclude (precision estimates aside) that the facility estimates and path criticality estimates are independent of the choice of input values for the seed and the number of pathfinding replications (providing this number is 117 or greater). This being the case, the estimates reflect

---

*For a given seed and MRUN, replications are identical.

OBSERVABLE: RANKING INDEX for path (1,20,14,8).

SOURCE	df	SS	MS	$F(\nu_1,\nu_2)$
SEED	2	0.0006	0.0003	0.92 (2,4)
MRUN	2	0.0007	0.0004	1.08 (2,4)
ERROR	4	0.0013	0.0003	
TOTAL	8	0.0026		

OBSERVABLE: Lower Bound (LB)

SOURCE	df	SS	MS	$F(\nu_1,\nu_2)$
SEED	2	0.0002	0.0001	0.36 (2,4)
MRUN	2	0	0	0
ERROR	4	0.0011	0.0003	
TOTAL	8	0.0013		

OBSERVABLE: Upper Bound (MIN or MINIMUM)

SOURCE	df	SS	MS	$F(\nu_1,\nu_2)$
SEED	2	0.0004	0.0002	2.67 (2,4)
MRUN	2	0	0	0
ERROR	4	0.0003	0.0001	
TOTAL	8	0.0007		

NOTE:
df = degrees of freedom; SS = sum of squares, MS = mean square

Figure 21. ANOVA Tables for Testing the Hypothesis that the Observables Are Independent of the Seed and the Number of Pathfinding Replications ($\geq$ 117)

the properties of the facility and the properties that are ascribed to it by the user rather than the incidental properties of the simulation.

Summary of Examples and Discussion -- An exhaustive set of examples of outputs for simulations in which detection probabilities are changed (from default values), delay and response times are varied, and different prior probability distributions are specified is beyond the scope of this document. The user will become familiar with the effects of these variations during applications of the PATHS procedure. By making use of the examples and discussion of this subsection, however, a summary of considerations useful to the user may be provided as follows:

1. The recommended minimum value for MRUN is 117; for IRUN, it is 1000.
2. Other factors being constant, improving facility security or increasing MRUN gives greater precison and less indeterminancy to the [LB, MIN] interval.
3. When LB > MIN, comparisons with a threshold, P_T, must assume that LB equals MIN. To actually bring about LB $\leq$ MIN, first change the seed.* Changing the seed, increasing MRUN, or both leads to LB $\leq$ MIN.
4. The number of paths generated is dependent upon the seed, MRUN, detection probabilities and response times, number of targets, and distribution types; it is therefore virtually unpredictable.
5. Ranking index alone is not a reliable indicator of path criticality (unless linear correlation is excellent).
6. Standard content (ranking indices and unconditional estimates) normally provides sufficient detail for interpretation of the output, particularly where linear correlation is good.
7. Multiple target simulations should be carried out first. Generally, such simulations will produce results which are predictable from many single target simulations.
8. Except for precision, estimates are independent of the choice of the values for the seed and MRUN $\geq$ 117 (and IRUN $\geq$ 1000).

---

*Central Processor (CP) time is unaffected by changes in the seed, but increasing MRUN increases the CP time.

5. OPERATIONAL CONSIDERATIONS

Run Times

Most simulations carried out by the PATHS procedure can be run in less than a minute and very often in a matter of seconds. The "run time" for a simulation is measured in terms of the number of Central Processor (CP) seconds that is required for the execution of the procedure. Run time does not include the time for input and output operations or the time spent in a queue when computer operations are done in a batch mode. Consequently, actual time will exceed run time. Nonetheless, because run time is indicative of the actual time and is the quantity which is most often used to determine the cost of computation, it is useful to understand (even qualitatively) the relation between the input to a simulation and the time for execution.

It is reasonable to expect that run time be a function of facility size (number of nodes and arcs), the number of Dijkstra searches per target (MRUN+1 for the interruption measure and MRUN for the time measure), the number of targets to be examined (N_T), the response times for each target (in effect, some fraction searched, f_{RT}, of all nodes that is determined by response time), the number of paths saved (N_p), and the number of Phase 2 replications per path saved (1 for the time measure, and IRUN for the interruption measure). As a matter of experience, the run time shows (1) a greater contribution from each of the Phase 1 $N_T \cdot$ MRUN searches than from each of the N_p or $N_p \cdot$ IRUN Phase 2 replications and (2) as much, if not more, dependence (under interruption) on the locations and response times for targets than on facility size.

Item (1), the greater contribution to run time per Phase 1 pathfinding replication in comparison to the Phase 2 contribution per replication/path, can be understood to result from the greater time needed to search for and identify a path than to carry out a single evaluation replication for a known path. Because only the nodes (and arcs) for known paths are treated, Phase 2 is essentially independent of facility size. Of course, as the number of paths generated (N_p) and the number of replications/path (IRUN) becomes large, the total Phase 2 contribution can become large. In Table 1, two facilities, one with 110 nodes and the other with 282 nodes, are considered.* Run times for IRUN = 0 (column 5, row 10; column 8, rows 1, 4, and 7) are subtracted from other values in the same set (MRUN = 1,10,50,100) and divided by the appropriate $L = N_p \cdot$ IRUN product to obtain estimates of run time per replication/path. It is reasonable to assume that, as L increases, these estimates converge to a value that represents the true value; therefore, an approximate value equal to or less than 1 ms is inferred. Note that for the run time of row 13, column 6, about 30% is contributed from Phase 2.

Item 2, the run time dependence on target location and response time in Phase 1, occurs because, while there is only one least detection search under the interruption measure and no least detection search under the time measure from off-site to all nodes, each of the MRUN searches from target to locus usually involves (unless the measure is time) just a fraction (effectively, f_{RT}) of all the facility nodes. Table 2 indicates that run time is nearly proportional to the $K = N_T \cdot$ MRUN product. This can be seen under each measure by comparing columns 7 and 12 to column 3. When time is the measure, the MRUN Phase 1 searches involve all nodes ($f_{RT} = 1$) and normalized run time (NORM) is approximated by the $N_T \cdot$ MRUN product; but, where most searches (all but one) involve some fraction of all the nodes (measure = interruption), normalized run time is less than the $N_T \cdot$ MRUN product would suggest.

Under the time measure, observed run times (OBS) include the Phase 2 (evaluation) contributions ($N_p \cdot$ time/path). Because there is only one replication per path and the time/path ≤ 0.001 second (from Table 1 on the assumption that time/path is independent of measure), this contribution is regarded as negligible.** Furthermore, observations under the interruption measure do not include contributions from Phase 2 because, with IRUN = 0, no evaluations are made. As a result, no adjustments are made for Phase 2 contributions to the observed values. For

---

[*] Simulations carried out for this table and subsequent tables use distributions (3,3,3) and SEED = 3719.

[**] This conclusion is also supported by the following computations using Table 3 (page 97, column 3):

$(8.54 - 1.06)/4 \cdot 2500 = 0.0007$ s/path

$(27.43 - 7.06)/10 \cdot 2500 = 0.0008$ s/path

Therefore, the largest contribution in Table 2 would be 0.01 second.

Table 1

Run Times (Seconds) for Two Facilities and the Product $L = N_p \cdot IRUN$ for the Interruption Measure Using Response Time = 1000 Minutes and Targets 1 (Facility 1) and 278 (Facility 2)

Column					Facility 1 (110 nodes)			Facility 2 (282 nodes)		
Row	1	2	3	4	5*	6*	7*	8*	9*	10*
	MRUN	IRUN	N_p	L			Cols. (6-5)/4			Cols. (9-8)/4
1	1	0	1	0				2.93		
2		50	1	50					3.06	0.003
3		100	1	100					3.11	0.002
4	10	0	1	0				8.93		
5		50	1	50					9.28	0.007
6		100	1	100					9.24	0.003
7	50	0	4	0				33.42		
8		50	4	200					34.87	0.007
9		100	4	400					34.93	0.004
10	100	0	9	0	20.15					
11		10	9	90		20.43	0.003			
12		100	9	900		22.07	0.002			
13		1000	9	9000		28.79	0.001			

Inferred value ≤ 0.001 seconds for evaluation time/path saved

*Run times in these columns are given in CP seconds

Table 2

Comparison of Run Times (Seconds) to K = N_T • MRUN for Interruption and Time Measures Using Facility 1 and Targets 1 and 1 Plus 3

Column				Time Measure (T)				Interruption Measure (I)[a]				
Row	1	2	3	4	5	6	7	8	9	10	11	12
	N_T	MRUN	K	OBS(N_p)	AJD(T)[b]	AJD(T)/(K-1)[c]	NORM(T)[d]	OBS (N_p)	AJD(I)[b]	AJD(I)/K[c]	RATIO[e]	NORM(I)[d]
1	1	1	1	0.45 (1)								
2	1	10	10	2.13 (7)	1.68	0.19[f]	9	0.79 (2)	0.34	0.034[f]	0.18	2
3	2	10	20	3.71 (4)	3.26	0.17[f]	18	1.08 (4)	0.63	0.032[f]	0.19	4
4	1	50	50	9.65 (8)	9.20	0.19	51	2.18 (3)	1.73	0.035	0.19	10
5	2	50	100	17.54 (12)	17.09	0.17	95	3.80 (8)	3.35	0.034	0.20	20
6	1	100	100	18.95 (9)	18.50	0.19	103	4.09 (5)	3.64	0.036	0.19	20
7	2	100	200	34.85 (16)	34.40	0.17	191	7.06 (10)	6.61	0.033	0.19	37
				Average time/search =		0.18		Average time/search =		0.034		

[a] For the interruption measure, IRUN is set equal to zero and response times (RT) = 3 minutes.

[b] The adjusted time (AJD) is obtained from the observed time (OBS) by subtracting 0.45 (row 1, col. 4). This is a (0.45) "set-up" time plus the time for one complete search (MNDPT1) from target to all nodes. Set-up time is assumed to be the same for all simulations and search time (MNDPT1) is assumed to equal the search time (PDIJK).

[c] For the time measure, K-1 is used because one search has been subtracted; for the interruption measure, K is used because the search (PDIJK), assumed equal to search (MNDPT1) = 0.45, has been subtracted.

[d] Normalized time NORM(T) = AJD(T)/(average time/search) and normalized time NORM(I) = RATIO • AJD(I)/(average time/search).

[e] RATIO = AJD(I) • (K-1)/AJD(T) • K, an indication of the effective fraction searched of all nodes in the facility.

[f] This alternation in values may be real; for (T), because the N_T loop is inside the MRUN loop (see PATHS listing, Appendix I); for (I), because of the preceding and/or because the RT = 3 locus for Target 3 encompasses fewer nodes than for target 1 (effect of location).

Table 3

Comparison of Run Times (Seconds) for Two Facilities Using the Interruption Measure and Response Times = 3 Minutes

			Facility 1 (110 nodes)				Facility 2 (282 nodes)			
Column Row	1	2	3*	4	5*	6	7*	8	9*	10
	IRUN	MRUN	Targets 1 & 3	N_p	Targets 1 & 201	N_p	Targets 278 & 315	N_p	Targets 306 & 636	N_p
1	0	10	1.06	4	0.82	2	3.09	1	2.40	1
2	0	100	7.06	10	5.86	6	16.09	2	7.85	4
3	2500	10	8.54	4	4.19	2	4.94	1	-	-
4	2500	100	27.43	10	16.58	6	20.85	2	-	-

*Run times in these columns are given in CP seconds.

both measures, some adjustment is required to remove the variation (with $K = N_T \cdot MRUN$) of the observed times owing to differences in the relative contributions of certain fixed components of the run times. Under time (measure), there is a contribution from a "set-up" time, and under interruption there is, in addition to "set-up" time, a contribution from one complete search (associated with PDIJK) from off-site to all nodes. The "set-up" time is a sum of the contributions to the CP time that arise from all manipulations other than Phase 1 searches and Phase 2 evaluations; this time is assumed to be a constant for all simulations. The times for searches from off-site to all nodes and from a target to a locus at RT = BIG are assumed to be equal; that is, time for a search (PDIJK, interruption) = time for a search (MNDPT1, time). An average* run time (row 1, column 4) for time simulations using one target and one pathfinding replication is an estimate of the "set-up" time plus the time for one search over all nodes. Adjusted values (AJD(T) and AJD(I), columns 5 and 9) are obtained from observed values (columns 4 and 8) by subtracting 0.45. From these, estimates (cols 6 and 10) of the average time/search--0.18 for time (measure) and 0.034 for interruption--are calculated. The ratio of these estimates (RATIO, Column 11) is, under the interruption measure, an estimate of the effect of response time on run time and an indication of the effective fraction (f_{RT}) of the total number of nodes (110) that is searched. For these targets (1 and 3) and response times of 3 minutes, a search apparently requires about 20% of the time that is needed to search all nodes (response time = BIG).

The combined effects of target location and response time can be seen in Table 3. Depending upon target location, a given response time locus can circumscribe a varying number of nodes. Therefore, it is difficult to consider the effects of location and response time separately. Comparison of the run times of columns 3 and 5 or columns 7 and 9 for IRUN = 0 shows the differences within facilities as the specification of targets is changed. This difference is not as pronounced for facility 1 as it is for facility 2. For facility 2, targets 278 and 315 (column 7) require significantly longer run times than targets 306 and 636 (column 9), especially for MRUN = 100. In addition, the differences between facilities are not as large as might be expected on the basis of facility size. It is, in fact, reasonable to expect the run time to depend upon the square of the number of facility nodes (see Reference 2, p. 16 and Reference 10, p. 424). No such relationship is clear in

---

*Average for three identical simulations.

Table 3, however, and, indeed, the run times for targets 306 and 636 of facility 2 (282 nodes) are comparable to those for targets 1 and 3 of facility 1 (110 nodes). This does not mean that run time is independent of facility size but simply that other effects can dominate when interruption is the measure. In fact, when IRUN = 2500, Table 3 shows that the Phase 2 (N_p · IRUN) contributions are particularly significant; so much so that, without knowing beforehand the number of facility nodes, facility 1 might be assumed to be larger than facility 2.

There is no certain means of predicting the run time for a simulation. The number of paths generated and the average time/search cannot be known in advance. Reasonable estimates of run time can only be made based upon the results of previous simulations. Thus, if the time for an interruption simulation in which MRUN = 117 and IRUN = 1000 is known, the time for the same simulation using MRUN = 234 and IRUN = 2500 can be estimated (subject to an unforseeable change in N_p). Similar estimates can be made for time simulations based upon relative values of MRUN. Normally, an overestimate of the Phase 1 contribution to an interruption simulation can be obtained as the run time for the corresponding time simulation (inputs are identical, except response times and response time distributions are defaulted to BIG and 1).

Storage Requirements

Facility size will determine the storage requirements for the PATHS procedure. The PATHS listing (Appendix I), provided as an example at the end of this guide, includes dimension statements that determine the maximum storage space allocable to this example of the procedure. From the listing it is apparent that the largest facility that can be treated, without altering the storage limits, consists of 138 nodes and 515 arcs with, at most, 20 targets (terminals). For a larger facility then, or for one with a greater number of targets, more storage space is required. Although the necessity for changing the storage space is remote (the procedure is provided with storage appropriate to the application), the user should be aware of these considerations.

BLANK PAGE

6. CONCLUSIONS

PATHS is one of a number of probabilistic models that are available for treating the stochastic shortest route problem. This problem arises in a number of contexts, but this user's guide focuses on the application of PATHS to the analysis of the effectiveness of facility safeguards. In this context, PATHS is more comprehensive than other models of similar purpose because the estimates of effectiveness that it provides take account of the uncertainty in the performance of safeguards through the use of random variables, the values of which are obtained by random sampling (Monte Carlo simulation) of prior probability distributions. Unfortunately, it is also in this context that the tenability of PATHS and competing models cannot be tested.

The tenability of a model is most often construed in terms of its predictive value and, in this respect, there is no way to determine how well PATHS works without having for comparision a real process which is an accurate reflection of probability as relative frequency. Conceptually such a process might have taken the form of a series of experiments in which an individual, in the role of an intruder, repeatedly traverses each and every path available within a facility, and the number of instances he is interrupted and the circumstances of these interruptions (guard response time, traversal times, triggering of detectors) are noted. For any but the most elementary facility and narrow range of circumstances, these experiments are impractical. Indeed, PATHS and related models were developed in order to make such experiments unnecessary.

Ultimately, the case for PATHS rests upon its completeness and accuracy of representation, its internal consistency, the reasonableness of the estimates obtained through its use, and the agreement of these estimates (taking into account inherent differences in treatment) with those obtained from related models. With respect to these attributes then, PATHS may be regarded as a satisfactory model.

BLANK PAGE

REFERENCES

[1] L. D. Chapman et al., *SAFE Users Manual Volume I: Introduction to SAFE*, SAND79-2247, NUREG/CR-1246 (Albuquerque: Sandia National Laboratories, 1980).

[2] B. L. Hulme, *MINDPT: A Code for Minimizing Detection Probability Up to a Given Time Away from a Sabotage Target*, SAND77-2039 (Albuquerque: Sandia Laboratories, December 1977).

[3] H. A. Bennett, *User's Guide for Evaluating Physical Security Capabilities of Nuclear Facilities by the EASI Method*, SAND77-0082, NUREG-0184 (Albuquerque: Sandia Laboratories, June 1977).

[4] H. A. Bennett, *The "EASI" Approach to Physical Security Evaluation*, SAND76-0500 (Albuquerque: Sandia Laboratories, January 1977).

[5] R. B. Worrell and D. W. Stock, *A SETS User's Manual for the Fault Tree Analyst*, SAND77-2051, NUREG/CR-0465 (Albuquerque: Sandia Laboratories, November 1978).

[6] B. L. Hulme, *Graph Theoretic Models of Theft Problems. I. The Basic Theft Model*, SAND75-0595 (Albuquerque: Sandia Laboratories, reprinted April 1977).

[7] B. L. Hulme, *Pathfinding in Graph-Theoretic Sabotage Models: I. Simultaneous Attack by Several Teams*, SAND76-0314 (Albuquerque: Sandia Laboratories, July 1976).

[8] R. C. Hall and R. D. Jones, *A Scientific Data Base for Safeguards Components*, SAND78-1766, NUREG/CR-0459 (Albuquerque: Sandia Laboratories, December 1978).

[9] E. W. Dijkstra, "A Note on Two Problems in Connextion with Graphs," *Numer. Math.*, *1* (1959), 269-271.

[10] J. Y. Yen, "Finding the Lengths of All Shortest Paths in N-Node Nonnegative-Distance Complete Networks Using $\frac{1}{2}N^3$ Additions and N^3 Comparisons," *J. Assoc. Comput. Mach.* *19*, (1972), 423-424.

[11] T. A. Williams and G. P. White, "A Note on Yen's Algorithm for Finding the Length of all Shortest Paths in N-Node Nonnegative-Distance Networks," *J. Assoc. Comput. Mach.* *20*, (1973), 389-390.

[12] *FORTRAN Extended Version IV Reference Manual*, Control Data Corporation, 1975.

[13] C. R. Hicks, *Fundamental Concepts in the Design of Experiments*, 2nd ed., (New York: Holt, Rinehart and Winston, Inc., 1973).

BLANK PAGE

APPENDICES

BLANK PAGE

APPENDIX A - Expected Values and Variance of the Interruption Probabilities

Appendix A is provided in order to demonstrate (1) that given the means or expected values of the detection probability prior distributions, then the expected values of the interruption probabilities (and hence the estimates of these values) are dependent on only the expected values of the prior distributions of the time-dependent variables and (2) that when the expected values for the detection probabilities are fixed, greater precision is obtained for the variance of the interruption probabilities (and the estimates of the variance).

Let D denote the set, {d}, of all node and arc detection probabilities and let T denote the set, {t}, of all node delay times, arc delay times, and guard response times. Assume that the members of D are independent random variables and that the members of T are independent random variables. Furthermore, assume that the members of T and D are independent of each other; that is, the events $T = t$ and $D = d$ are independent events for all d and t where d and t jointly exist.

For any path in the facility, let the random variable, $P(D,T)$, be the interruption probability for the path. In Phases 1 and 2 of the PATHS treatment, the expected value $E_{T,D}[P(D,T)]$, is computed by making use of the general (D and T need not be independent) relation:

$$E_{T,D}[P(D,T)] = E_T\{E_D[P(D,T|T)]\} \qquad \text{(A-1)}$$

where $E_{T,D}$ is the expectation taken first with regard to detection probability and then with regard to time. For a given target, the conditional probability of interruption, P_c, is sought in each Phase 1 run by examining all possible paths. $P_c = E_D[P(D,T|T)]$ is recorded only for the path that minimizes $E_D[P(D,T|T)]$. $E_T(P_c)$ is then estimated by the statistic which is the arithmetic mean of a randomly generated sample consisting of the values of $E_D[P(D,T|T)]$ that are least in each run. In the Phase 2 runs, the unconditional probability of interruption, P_u, for each path selected in Phase 1 is evaluated in every run. $E_T(P_u)$ for each path is estimated by the statistic which is the arithmetic mean of a random sample consisting of the computed values, $P_u = E_D[P(D,T|T)]$, that are obtained from all runs without regard to minimization of $E_D[P(D,T|T)]$.

Having noted that

$$E_T(P_c) = E_{T,D}[P(D,T)]_{\text{COND}} = E_T \{\min E_D[P(D,T|T)]\}, \qquad \text{(A-2)}$$

and

$$E_T(P_u) = E_{T,D}[P(D,T)]_{\text{UNCOND}} = E_T \{E_D[P(D,T|T)]\} \qquad \text{(A-3)}$$

for generated paths, $E_{T,D}[P(D,T)]$ and $\text{Var}[P(D,T)]$ are considered making use of Eq. (A-1), the independence conditions, and the definition of the interruption probability, P(D,T).

P(D,T) for all d and t has the following form:

$$P(d,t) = \sum_{i=1}^{n} H(t_i) d_i \prod_{j=0}^{i-1} (1-d_j) = \sum_{i=1}^{n} H(t_i) G(d_i) \qquad \text{(A-4)}$$

where $G(d_i) = d_i \prod_{j=0}^{i-1}(1-d_j)$, $d_0 = 0$, $H(t_i)$ is a function of the delay and response times, and the summation is over all n components (arcs and nodes) of the path. In the course of examining the facility's paths, three cases can arise depending upon the response time locus. This locus is determined by the magnitude of the response time, RT, which may vary from zero to t_{max}, a time that is large compared to the sum of all delay times for any path. These three cases are described in Table A-1.

For Case 1, the interruption probability is not meaningful because the guard response is so tardy that interruption is not possible. By default, the measure is time and minimum traversal times are computed for the selected paths. For Case 2, the measure is solely detection probability, and the computation of minimum detection probabilities for the selected paths is a special instance of Case 3.

Table A-1

Facility Cases

Case	Response Time	Index	$H(t_i)$	$G(d_i)$	$P(d,t)$
1	$RT \geq t_{max}$	$i=1,n$	t_i	*	not defined**
2	$RT = 0$	$i=1,n$	*	$G(d_i)$	$\sum_{i=1}^{n} G(d_i)$
3	$0<RT<t_{max}$	$i<k$	*	$G(d_i)$	$\sum_{i=1}^{k-1} G(d_i) + H(t_k)G(d_k)$
		$i=k$	$H(t_k)^{\dagger}$	$G(d_k)$	
		$k<i\leq n$	t_i	*	

*Not defined

**$T = \sum_{i=1}^{n} H(t_i) = \sum_{i=1}^{n} t_i \leq t_{max}$, where T is the cumulative traversal time.

†$H(t_k) = f_k$ is a number between zero and one.

Case 3 gives

$$P(d,t) = \sum_{i=1}^{k-1} G(d_i) + H(t_k)G(d_k) \qquad \text{(A-5)}$$

Because D and T are independent, the joint density function of D and T is

$$f(d,t) = f_1(d)f_2(t) \qquad \text{(A-6)}$$

Where $f_1(d)$ and $f_2(t)$ are the marginal probability functions of D and T, and in particular

$$f_1(d) = f_1(d_i) \qquad i=1,n \qquad \text{(A-7)}$$

where the $f_i(d_i)$ are the prior probability distributions for the detection weights associated with the n path components.

By definition

$$E_D\ [P(D,T|T])\ = \int_{-\infty}^{\infty} P(d,t)f(d|t)\delta d$$

$$= \int_{-\infty}^{\infty} P(d,t)\frac{f(d,t)}{f_2(t)}\delta d \qquad \text{(A-8)*}$$

$$= \int_{-\infty}^{\infty} P(d,t)f_1(d)\delta d$$

By making use of Eqs. (A-5), (A-7), and (A-8), the following equation can be obtained without loss of generality:

$$E_D[P(D,T|T)] = \int_{-\infty}^{\infty} \sum_{i=1}^{k-1} G(d_i)f_1(d_i)\delta d_i + H(t_k) \int_{-\infty}^{\infty} G(d_k)f_1(d_k)\delta d_k$$

$$= E\left[\sum_{i=1}^{k-1} G(d_i)\right] + H(t_k)E[G(d_k)] \qquad \text{(A-9)}$$

Again, without loss of generality:

$$E\left[\sum_{i=1}^{k-1} G(d_i)\right] = \sum_{i=1}^{k-1} E[G(d_i)] \qquad \text{(A-10)}$$

Consequently,

$$E_D[P(D,T|T)] = \sum_{i=1}^{k-1} E[G(d_i)] + H(t_k)E[G(d_k)] \qquad \text{(A-11)}$$

The members of D are independent and any term, $E[G(d_i)]$, in Eq. (A-11) is

---

*D and T are jointly continuous, the integrals are Riemann rather than Lebesgue-Stieltjes, and $f(d|t) = f(d,t)/f_2(t)$ for all t such that $f_2(t) > 0$.

$$E[G(d_i)] = E(d_i) \prod_{j=0}^{i-1} E(1-d_j)$$

$$= E(d_i) \prod_{j=0}^{i-1} [1-E(d_j)] \qquad \text{(A-12)}$$

$$= G[E(d_i)]$$

where

$$E(d_i) = \int_{-\infty}^{\infty} d_i f_1(d_i) \delta d_i$$

Then

$$E_D[P(D,T|T)] = \sum_{i=1}^{k-1} G[E(d_i)] + H(t_k)G[E(d_k)]$$

$$= P[E(D),T] \qquad \text{(A-13)}$$

Taking the expectation with respect to time of both sides of Eq. (A-13) and making use of Eq. (A-1), we have

$$E_T\{E_D[P(D,T|T)]\} = E_T\{P[E(D),T]\} \qquad \text{(A-14)}$$

and

$$E_{T,D}[P(D,T)] = E_T\{P[E(D),T]\} \qquad \text{(A-15)}$$

The relation, (Eq. A-15), demonstrates that, given the expected values of the detection probability prior distributions, the expected values of the conditional and unconditional probabilities of interruption, $E_T(P_c)$ and $E_T(P_u)$, depend only upon the expected values of the prior distributions of the time-dependent variables.

Because $E_{T,D}[P(D,T)]$ and E_T {[P[E(D),T]} have the same expectation, it is possible, and computationally simpler, to use values that are obtained from the distribution of $P[E(D),T]$ rather than from the distribution of $P(D,T)$ in the Phase 1 and Phase 2 random samples. Furthermore, if the prior probability distribution for the detection probabilities, $f_1(d)$, is deterministic and the values of $E(D)$ are fixed, the variance of $P(D,T)$ is less than that which obtains when the D are random. As a consequence, greater precision is obtained in the estimates of $E_T(P_c)$ and $E_T(P_u)$.

In general,

$$\mathrm{Var}[P(D,T)] = E_T\{\underset{T}{\mathrm{Var}}\ [P(D,T|T)]\} + \underset{T}{\mathrm{Var}}\{E_D[(P(D,T|T)]\} \qquad \text{(A-16)}$$

and taking the variance of both sides of Eq. (A-13), gives

$$\underset{T}{\mathrm{Var}}\{E_D[P(D,T|T)]\} = \underset{T}{\mathrm{Var}}\{P[E(D),T]\} \qquad \text{(A-17)}$$

As a result, with D and T random,

$$\mathrm{Var}[P(D,T)] = E_T\{\underset{T}{\mathrm{Var}}[P(D,T|T)]\} + \underset{T}{\mathrm{Var}}\{P[E(D),T]\} \qquad \text{(A-18)}$$

When $f_1(d)$ is deterministic and the values of $E(D)$ are fixed (this is the case for PATHS), then $\underset{T}{\mathrm{Var}}[P(D,T|T)]$ is zero and the variance of $P(D,T)$ is simply obtained as the variance of $P[E(D),T]$; that is,

$$\mathrm{Var}[P(D,T)] = \underset{T}{\mathrm{Var}}\{P[E(D),T]\} \qquad \text{(A-19)}$$

and it is apparent that

$$\mathrm{Var}[P(D,T)]_{D\ \text{fixed}} < \mathrm{Var}[P(D,T)]_{D\ \text{random}} \qquad \text{(A-20)}$$

It can, of course, be argued that if $f_1(d)$ is made deterministic, then the inherent randomness in the detection probabilities is no longer accounted for. This is a reasonable argument. The ultimate justification for making $f_1(d)$ deterministic in the PATHS treatment is an empirical one; namely, that by assigning other prior distributions to $f_1(d)$, no significant difference in the results is observed.

Because of Eqs. (A-15) and (A-19), those statistics that are estimates of $E_{T,D}[P(D,T)]$ and Var $[P(D,T)]$, the sample means and unbiased sample variances (or sample standard deviations), are computed from values of $P[E(D),T]$ where the prior distribution of D is deterministic. In so doing the time for computation is reduced and greater precision is obtained in the estimates. Moreover, because the samples means are, by the Central Limit Theorem, approximately normally distributed, the standard deviations of the sample means (given in the summary output of PATHS) approach zero and the sample means approach their expected values, $E_T(P_c)$ and $E_T(P_u)$, with probability one as the number of Phase 1 or Phase 2 runs (at least >30) is increased.

114

BLANK PAGE

APPENDIX B - Lower and Upper Bounds to the Minimum Interruption Probability

Appendix B demonstrates that the unknown parameter of the PATHS model, the minimum interruption probability, min P, is bounded by LB, the Phase 1 facility estimate or lower bound, and MIN, the Phase 2 facility estimate or upper bound.

Let {A} denote the set of all N physical paths in the facility and let $P(A_{nt})$ represent the value of the interruption probability for the n^{th} path on the t^{th} run (t^{th} realization of the time parameters). For each of the N paths in the facility, there is an expected value, $E_T[P(A_{nt})]$, over all Phase 1 runs. The minimum of these expected values over all N paths is the facility minimum, min P or

$$\min P = \min_{N} \{E_T[P(A_{nt})]\} \qquad \text{(B-1)}$$

Each path generated in Phase 1 is associated with a least value, $\min_{N} [P(A_{nt})]$, for each run in which the path is generated. The expected value over all generated paths in all Phase 1 runs is $E_T\{\min_{N}[P(A_{nt})]\}$

and

$$E_T\{\min_{N}[P(A_{nt})]\} \le \min_{N} \{E_T[P(A_{nt})]\} \qquad \text{(B-2)}$$

but $\min_{N} [P(A_{nt})]$ is the Phase 1 conditional probability of interruption, $P_c(A_{nt})$. Hence

$$E_T[P_c(A_{nt})] \le \min_{N}\{E_T[P(A_{nt})]\} \qquad \text{(B-3)}$$

or

$$E_T[P_c(A_{nt})] \le \min P \qquad \text{(B-4)}$$

The statistic, LB, which is an estimate of $E_T[P_c(A_{nt})]$, is the lower bound. LB is computed in Phase 1 by

$$LB = \sum_{t=1}^{MRUN} P_c(A_{nt})/MRUN \tag{B-5}$$

and therefore if MRUN (the Phase 1 sample size) is sufficiently large

$$LB \leq \min P \tag{B-6}$$

In Phase 2, the expected value over all runs for the interruption probability of each of the K paths ($K \subset N$) generated in Phase 1 is

$$E_T[P_u(A_{kt})] = E_T[P(A_{kt})] \tag{B-7}$$

The minimum of these expected values over all K paths is

$$\min_K P = \min_K \{E_T[P_u(A_{kt})]\} \tag{B-8}$$

If the path with interruption probability equal to min P is one of these paths, then

$$\min_K P = \min P \tag{B-9}$$

If it is not one of the K paths generated, then, by definition of min P

$$\min_K P > \min P$$

The statistic, MIN, which is an estimate of $\min_K P$, is the upper bound. MIN is computed in Phase 2 by

$$MIN = \min_K \left[\sum_{t=1}^{IRUN} P_u(A_{kt})/IRUN\right] \tag{B-11}$$

and therefore if IRUN (the Phase 2 sample size) is sufficiently large

$$MIN \geq \min P \tag{B-12}$$

Combining Eqs. (B-6) and (B-12) gives

$$LB \leq \min P \leq MIN \qquad (B\text{-}13)$$

and Eq. (B-13) provides the theoretical basis upon which inferences may be made concerning the security of the facility.

In the actual application of PATHS, the utility of Eq. (B-13) is strongly dependent upon the Phase 1 and Phase 2 sample sizes. The Phase 1 and Phase 2 runs are independent and statistically identical. Therefore all Phase 1 and Phase 2 statistics are, by the Central Limit Theorem, approximately normally distributed provided that the number of runs in each phase is sufficiently large. Given this, the estimates, LB and MIN, of the sample means will then be close to their expected values. The precision of these estimates may be inferred from the sample standard deviations and the standard deviations of the sample means. Both the absolute and relative numbers of Phase 1 and 2 runs should be selected so that the resulting standard deviations of the sample means are small, and LB $\leq$ MIN. A Phase 1 sample size that is too small relative to the Phase 2 sample size can lead to the generation of a value for the lower bound that is greater than that for the upper bound.

BLANK PAGE

APPENDIX C - Relationship between Conditional and Unconditional Interruption Probabilities

Appendix C demonstrates the relationship between the expected value for the *conditional* interruption probability (traversal time) that can be computed for a path in the path generation phase (Phase 1) and the expected value for the *unconditional* interruption probability (traversal time) that is computed for a path in the path evaluation phase (Phase 2).

Let A denote the event that a path is interrupted (or traversed), A_M denote the event that the path is the minimum path (most critical) with respect to the measure, and A_{NM} denote the event that the path is *not* the minimum path with respect to the measure. In general (but not in Phase 1), the event A can occur as the result of either of the two mutually exclusive events A_M and A_{NM}. The complement of $A_M = A_M'$ is A_{NM} and for any sets A and A_M:

$$\begin{aligned} A &= (A \cap A_M) \cup (A \cap A_M') \qquad \text{(C-1)} \\ &= (A \cap A_M) \cup (A \cap A_{NM}) \end{aligned}$$

but $(A \cap A_M)$ and $(A \cap A_{NM})$ are mutually exclusive because A_M and A_{NM} are. Hence, for a particular path, the interruption probability (or traversal time) is

$$\begin{aligned} P(A) &= P(A \cap A_M) + P(A \cap A_{NM}) \qquad \text{(C-2)} \\ &= P(A_m)P(A|A_M) + P(A_{NM})P(A|A_{NM}) \end{aligned}$$

and

$$P(A) = P(A_M)P(A|A_M) + [1 - P(A_M)]P(A|A_{NM}) \qquad \text{(C-3)}$$

because $P(A_{NM}) = P(A_M') = 1 - P(A_M)$.

Consider the probabilities (or traversal times) of Eq. (C-3) to be random variables. Assuming that the samples obtained in the MRUN replications of Phase 1 and the IRUN replications of Phase 2 are statistically identical, then by taking the expectations

$$E[P(A)] = E[P(A_M)]\ E[P(A|A_M)] + \{1 - E[P(A_M)]\}E[P(A|A_{NM})] \quad \text{(C-4)}$$

Now $E[P(A)]$ is the mean interruption probability (traversal time) for the path, $E[P(A_M)]$ is the probability that the path is most critical, $E[P(A|A_M)]$ is the mean interruption probability (traversal time) for the path given that it is most critical, $1 - E[P(A_M)]$ is the probability that the path is not most critical, and $E[P(A|A_{NM})]$ is the mean interruption probability (traversal time) for the path given that the path is not most critical.

Consequently, the following identifications can be made:

$E[P(A)]$ = Phase 2 mean unconditional probability (traversal time) $E_T(P_u)$ over all replications for the path

$E[P(A_M)]$ = Phase 1 rank for the path

$E[P(A|A_M)]$ = Phase 1 mean conditional probability (traversal time) $E_T(P_c)$ over all replications (equal in number to the count) that generated the path

and therefore the mean conditional probability given that the path is not most critical is

$$E[P(A|A_{NM})] = [E_T(P_u) - (\text{rank})E_T(P_c)]/(1 - \text{rank}) \qquad \text{(C-5)}$$

APPENDIX D - Maximum Variance of a Random Variable with Interval [0,1]

Appendix D demonstrates that if the expected value of a random variable with interval [0,1] is P, then the variance of the random variable is at most P(1-P).

Let X be a random variable with values between and including 0 and 1, and let f(x) be the probability density function for X. Let

$$E(X) = \int_{[0,1]} xf(x)dx = P \qquad \text{(D-1)}$$

where [0,1] corresponds to $0 \leq x \leq 1$. Now

$$E(X^2) = \int_{[0,1]} x^2 f(x)dx \qquad \text{(D-2)}$$

but

$$\int_{[0,1]} x^2 f(x)dx \leq \int_{[0,1]} xf(x)dx \qquad \text{(D-3)}$$

because x^2 on [0,1] is less than or equal to x on [0,1]. Because

$$\begin{aligned} \mathrm{Var}(X) &= E(X^2) - [E(X)]^2 \\ &= \int_{[0,1]} x^2 f(x)dx - P^2 \qquad \text{(D-4)} \end{aligned}$$

then

$$\mathrm{Var}(X) \leq \int_{[0,1]} xf(x)dx - P^2 = P(1-P) \qquad \text{(D-5)}$$

122

BLANK PAGE

APPENDIX E - Determination of the Minimum Number of Pathfinding Replications

Appendix E considers the minimum Phase 1 sample size, i.e., the number of Phase 1 pathfinding replications, or runs, that must be specified in order to be reasonably sure that (1) a critical path or (2) a number of critical paths are generated at least once.

Let the random variable, X_i, represent the number of times a minimum path, A_i, is generated in Phase 1, and assume that

$$P_1(X_i=1) = p_i \geq p \tag{E-1}$$

for any path A_i; that is, for one run the probability that path A_i is generated (is a minimum) is at least p. Given a value for p, the number of Phase 1 runs, m, that will ensure that path A_i is generated at least once with a probability, for example, of at least 0.95 can be calculated. This is expressed by

$$P_m(X_i \geq 1) \geq 0.95 \tag{E-2}$$

where $X_i \geq 1$ denotes the event that A_i generated at least once.

Now

$$P_m(X_i \geq 1) = 1 - P_m(X_i=0)$$

$$P_m(X_i=0) = \genfrac{}{}{0pt}{}{m}{0} p_i^0 \, q_i^m \tag{E-3}$$

$$= q_i^m$$

and therefore

$$P_m(X_i \geq 1) = 1 - q_i^m \tag{E-4}$$

Because $q_i = 1 - p_i$, then $q_i \leq (1 - p)$ when $p_i \geq p$, and

$$P_m(X_i \geq 1) \geq 1- (1-p)^m = 0.95 \qquad \text{(E-5)}$$

If $p = 0.05$, then

$$(0.95)^m = 0.05 \qquad \text{(E-6)}$$

and $m = 58$. Consequently, if the number of Phase 1 runs is at least 58, then path A_i, with a 0.05 probability of being the minimum, has at least a 0.95 probability of being generated at least once.

Suppose now that a number of paths, for instance, a set of $k < m$ paths, is to be generated at least once in m Phase 1 runs with probability ≥ 0.95. If again $p_i \geq p = 0.05$, the number of runs that is required to assure that this event occurs can also be calculated.

Let the event be denoted by X where

$$X = X_1 \cap X_2 \cap \ldots \cap X_k \qquad \text{(E-7)}$$

Repeated application of De Morgan's second law gives

$$X' = X_1' \cup X_2' \cup \ldots \cup X_k' \qquad \text{(E-8)}$$

where

$$X' = -X_i \qquad \text{(E-9)}$$

Then

$$P_m(X') \leq P_m(X_1') + P_m(X_2') + \ldots + P_m(X_k'), \qquad \text{(E-10)}$$

the inequality arising from the fact that the X_i' are not mutually exclusive. Now

$$P_m(X \geq 1) = 1 - P_m(X') \qquad \text{(E-11)}$$

and recall that

$$P_m(X_i') = P_m(X_i = 0) = q_i^m \qquad \text{(E-12)}$$

Consequently

$$P_m(X\geq 1) \geq 1 - [q_1^m + q_2^m + \ldots + q_k^m]$$

$$\geq 1 - [(1-p_1)^m + (1-p_2)^m + \ldots + (1-p_k)^m] \qquad (E\text{-}13)$$

The upper bound to the number of paths, k, in the set is determined by requiring that

$$\sum_{i=1}^{k} p_i \simeq 1 \qquad (E\text{-}14)$$

In general, it is true that

$$\sum_{i=1}^{N} p_i = 1 \qquad (E\text{-}15)$$

where N = total number of physical paths. Thus

$$\sum_{i=1}^{k} p_i + \sum_{i=k+1}^{N} p_i = 1 \qquad (E\text{-}16)$$

but, by assuming that

$$\sum_{i=k+1}^{N} p_i = \varepsilon \qquad (E\text{-}17)$$

where ε is arbitrarily close to zero, Eq. (E-14) is obtained. This is equivalent to insisting that only the paths in the set {k} have probabilities of being minimum paths that are significant. All other paths, the average probabilities of which are $\varepsilon/(N-k)$ of being minimum paths, are not significant.

When $p_i \geq P$

$$\sum_{i=1}^{k} P \lesssim 1 \qquad (E\text{-}18)$$

and therefore

$$k \leqq 1/p \qquad \text{(E-19)}$$

The largest set than can be generated, having specified the value of p and having required that $\varepsilon = 0$, is the set for which each $p_i = p$ or

$$k = 1/p \qquad \text{(E-20)}$$

In this instance, the p_i are uniformly distributed, and because this distribution can occur, the minimum acceptable value of m is determined by this worst-case distribution. This can be seen as follows.

If it is required that the right-hand side of (E-13) be equal to a specified probability regardless of the distribution of the p_i, then

$$\sum_{i=1}^{k'} (1-p_i)^{m'} = k(1-p)^{m} \qquad \text{(E-21)}$$

Each term, $(1-p_i)$, is less than $(1-p)$ because $p_i > p$. Furthermore, k' must be less than k, and therefore (E-21) can hold only if $m' < m$. Because the instance (the uniform distribution) for which the right-hand side of (E-13) is $k(1-p)^m$ must be accounted for, the minimum acceptable number of runs is $m > m'$ and the requirement that $P_m(X \geq 1) \geq 0.95$ gives

$$P_m(X \geq 1) \geq 1 - k(1-p)^m = 0.95 \qquad \text{(E-22)}$$

With $k = 1/p$

$$1 - \frac{1}{p}(1-p)^m = 0.95 \qquad \text{(E-23)}$$

or

$$m = \ln(0.05p)/\ln(1-p) \qquad \text{(E-24)}$$

which gives $m = 117$ for $p = 0.05$

Thus, by merely doubling the number of Phase 1 runs, there is at least a probability of 0.95 that the k members of the set will be generated one or more times providing each path of the set has at least a probability of 0.05 of being the minimum interruption path. Obviously if the members of the set have probabilities greater than 0.05, the set becomes more exclusive and, given the same number of runs (117), the probability with which the set is generated becomes greater. Indeed, if $p_i = p = 0.5$, the set of two members is generated with a probability that is very nearly 1.

BLANK PAGE

APPENDIX F - Descriptions of Prior Probability Distributions Used in PATHS

Appendix F provides descriptions of the five prior probability distributions in terms of the quantities that characterize each distribution. For the uniform and triangular distributions, these quantities are the mode, minimum (min), and maximum (max); for the normal and lognormal distributions, these quantities are the mean and standard deviation. Each time parameter to which a distribution has been assigned is therefore associated with a set of values for the quantities that characterize the distribution. The PATHS subroutines, DFLT and MOD, are the agents by which input values for the time parameters are converted to the quantities that characterize the distributions. In DFLT, each single input value (obtained from Tape 20 which contains the facility graph data) is equated to one of these characterizing quantities and the values of other characterizing quantities in the set are computed from it. In MOD, the values for the quantities in a characterizing set are either obtained from DFLT (examine distribution) or they are specified anew by the user (modify distribution). With the exception of the lognormal distribution, MOD accepts the new values as entered, and they are not conditioned to yield symmetrical distributions or specific coefficients of variation.

Let y equal the input value to DFLT that is associated with the time parameter t. In DFLT, y is regarded as the mode (most probable value) for the distribution that is assigned to the time parameter. With the exception of the lognormal distribution, the means or expected values for the distributions are equal to the modes; that is, the distributions are not skewed. For both the normal and lognormal distributions, the standard deviations are set equal to two-tenths of the means. Where min and max apply, they are set equal to 1/2 and 3/2 of the mode.

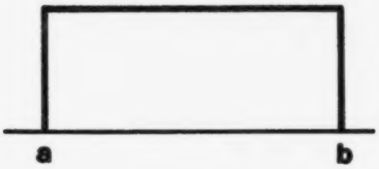
Quantities that characterize the distributions are labeled (a, b, c, μ, σ,) in the accompanying figures.

Deterministic Distribution

mode = mean = y

Uniform Distribution

$$f(t) = \begin{cases} 1/(b-a), & a \le t \le b \\ 0, & \text{elsewhere} \end{cases}$$



mode = mean = y

min = a = 0.5y

max = b = 1.5y

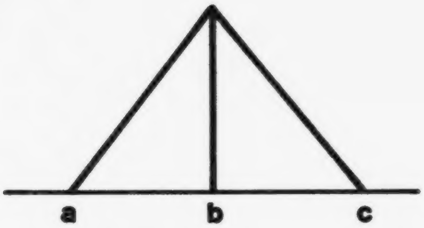
mean = (a+b)/2 = y

standard deviation = (b-a)/ 12 = $y/(2\sqrt{3})$

coeff. of variation = $1/(2\sqrt{3})$ = 0.29

Triangular Distribution

$$f(t) = \begin{cases} \frac{2(t-a)}{(b-a)(c-a)}, & a \le t < b \\ \frac{2(c-t)}{(c-b)(c-a)}, & b \le t \le c \\ 0, & \text{elsewhere} \end{cases}$$



mode = b = y

min = a = 0.5y

max = c = 1.5y

mean = (a+b+c)/3 = y

$$\text{standard deviation} = {\frac{a(a-b) + c(c-a) + b(b-c)}{18}}^{1/2} = y/(2\sqrt{6})$$

coefficient of variation = $1/(2\sqrt{6})$ = 0.20

Normal Distribution

$$f(t) = \frac{1}{\sigma\sqrt{2\pi}}\, e^{-(t-\mu)^2/2\sigma^2}, \quad -\infty < t < \infty$$

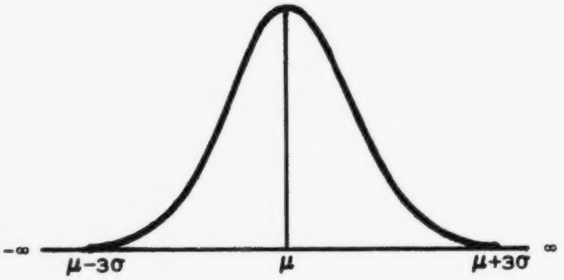
where $\sigma > 0$ and $-\infty < \mu < \infty$

mode = μ = y

mean = μ = y

standard deviation = σ = 0.2y

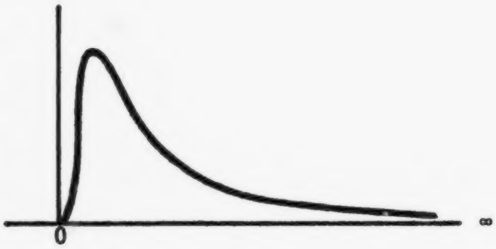
coefficent of variation = σ/μ = 0.20



Lognormal Distribution

Let μ = mean of the normal distribution and let σ = standard deviation of the normal distribution. For the lognormal distribution then

$$f(t) = \frac{t^{-1}}{\sigma\sqrt{2\pi}} e^{-(\ln t-\mu)^2/2\sigma^2}, \; 0 < t < \infty$$



where $\sigma > 0$ and $0 \leq \mu < \infty$

mode = $e^{(\mu-\sigma^2)} = y$

mean = $e^{(2\mu+\sigma^2)/2} = ye^{3\sigma^2/2}$

standard deviation = $e^{(2\mu+\sigma^2)/2} \cdot (e^{\sigma^2} - 1)^{1/2} = ye^{3\sigma^2/2}(e^{\sigma^2} - 1)^{1/2}$

coefficient of variation = $(e^{\sigma^2} - 1)^{1/2} \triangleq 0.20$

Hence, $\sigma^2 = 0.039$, $\mu = \ln y + \sigma^2$, and mean = 1.06y.

In MOD, (input) values for the various characterizing quantities are already known or specified anew, and they are not evaluated from the modes as in DFLT. Consequently, the expressions for the mean and standard deviation, for each distribution described previously, are obtained in MOD by ignoring the terms that contain y. The coefficient of variation for each distribution is then, in general, the ratio of the standard deviation to the mean. The only constraint imposed by MOD is that the coefficient of variation remain equal to 0.20 for the lognormal distribution.

BLANK PAGE

APPENDIX G - Cumulative Probability of Detection for a Path

Appendix G derives the expression for the cumulative probability of detection for a path extending from a boundary up to but not including a given node in terms of the detection probability weights for the path components (nodes and arcs).

Let D denote the event that detection of the adversary occurs as a result of one of the mutually exclusive events $D_1, D_2, \ldots, D_n$. D_i is the event that detection occurs at path component i, and $P(D_i)$ is the probability of this event. If all the events up to but not including a terminal node i + 1 are considered, then

$$\begin{aligned} D &= D_1 \cup D_2 \cup \cdots \cup D_i \\ &= \underline{D} \cup D_i \end{aligned} \tag{G-1}$$

where $\underline{D}$ is the union of the events preceding the event D_i.

The event D_i can be partitioned so that

$$D_i = (D_i \cap \underline{D}) \cup (D_i \cap \underline{D}') \tag{G-2}$$

where $\underline{D}'$ is the complement of $\underline{D}$. Now

$$(D_i \cap \underline{D}) = \emptyset \tag{G-3}$$

because D_i and $\underline{D}$ are mutually exclusive. As a result

$$\begin{aligned} D_i &= \emptyset \cup (D_i \cap \underline{D}') \\ &= (D_i \cap \underline{D}') \end{aligned} \tag{G-4}$$

and

$$P(D_i) = P(\underline{D}')P(D_i | \underline{D}') \tag{G-5}$$

Now $P(D_i|\underline{D}')$ is d_i, the detection probability weighting for component i and since, by De Morgan's first law,

$$\underline{D}' = D_1' \cap D_2' \cap \ldots \cap D_{i-1}' \tag{G-6}$$

then

$$P(\underline{D}') = P(D_1' \cap D_2' \cap \ldots \cap D_{i-1}')$$

$$= P(D_1')P(D_2'|D_1')\ldots P(D_{i-1}'|D_1' \cap D_2' \cap \ldots \cap D_{i-2}') \tag{G-7}$$

but in general

$$P(D_i') = P(D_i'|D_1' \cap D_2' \cap \ldots \cap D_{i-1}') \tag{G-8}$$

and therefore

$$P(\underline{D}') = P(D_1')P(D_2')\ldots P(D_{i-1}') \tag{G-9}$$

$P(D_j')$ is $1-d_j$, the non-detection probability weighting for component j. Thus

$$P(\underline{D}') = (1-d_1)(1-d_2)\ldots(1-d_{i-1})$$

$$= \prod_{j=1}^{i-1} (1-d_j) \tag{G-10}$$

$$= \prod_{j=0}^{i-1} (1-d_j)$$

where $d_0 = 0$. Consequently, using Eq. (G-5)

$$P(D_i) = d_i \prod_{j=0}^{i-1} (1-d_j) \tag{G-11}$$

From Eq. (G-1) and the fact that the events $D_1, D_2, \ldots, D_i$ are mutually exclusive

$$P(D) = P(D_1) + P(D_2) + \ldots + P(D_n)$$

$$= \sum_{i=1}^{n} d_i \prod_{j=0}^{i-1} (1-d_j) \tag{G-12}$$

P(D) is the cumulative detection probability up to but not including the n+1 th component. Redefining P(D) as P_{n+1} gives

$$P_{n+1} = \sum_{i=1}^{n} d_i \prod_{j=0}^{i-1} (1-d_j) \tag{G-13}$$

where $d_0 = 0$, n is even and refers to the component (an arc) preceding the n+1th node. Odd i refer to nodes, and even i refer to arcs.

BLANK PAGE

APPENDIX H - Monte Carlo Sampling of Prior Probability Distributions

Appendix H describes the Monte Carlo method for sampling the prior probability distributions to obtain random values (for the time parameters) that are used in the Phase 1 (path generation) sample and the Phase 2 (path evaluation) sample.

Let $\{RN_i\}$ be a set of random deviates (numbers) from a uniform distribution on the interval [0,1] with mean μ_r and variance σ_r^2.

Let the random variable X from a uniform distribution have values min = a and max = b. It is desired to generate a value, x, for the variable X such that

$$P(a \leq X \leq x) = RN_i \qquad a \leq x \leq b \qquad \text{(H-1)}$$

Because

$$P(a \leq X \leq x) = \int_a^x dt/(b-a) \qquad a \leq x < b \qquad \text{(H-2)}$$

$$= 0 \qquad \text{elsewhere}$$

the randomly generated value of X may be computed from

$$x = a + (b-a)RN_i \qquad a \leq x \leq b \qquad \text{(H-3)}$$

If the variable X is triangularly distributed with min = a, mode = b, and max = c, a value for X such that

$$P\ (a \leq X \leq x) = RN_i \qquad a \leq x \leq b \qquad \text{(H-4)}$$

may be obtained by using

$$P\ (a \leq X \leq x) = \int_a^x 2(t-a)dt/(b-a)(c-a) \qquad a \leq x \leq b \qquad \text{(H-5)}$$

from which

$$x = a + [(b-a)(c-a)RN_i]^{1/2} \qquad a \leq x \leq b \qquad (H-6)$$

Now

$$P(a \leq X \leq x) + P(x \leq X \leq c) = 1 \qquad a \leq x \leq c \qquad (H-7)$$

and therefore using Eq. (H-5)

$$P(x \leq X \leq c) = 1 - RN_i \qquad b \leq x \leq c \qquad (H-8)$$

Because

$$P(x \leq X \leq c) = \int_x^c 2(c-t)dt/(c-b)(c-a) \qquad b \leq x \leq c \qquad (H-9)$$

then

$$x = c - [(c-b)(c-a)(1-RN_i)]^{1/2} \qquad b \leq x \leq c \qquad (H-10)$$

Hence, Eqs. (H-6) and (H-10) may be used to compute randomly generated values of X for the triangular distribution.

Consider now a random variable X with finite mean μ and variance σ^2. The associated standardized random variable with mean 0 and variance 1 is

$$V = \frac{X - \mu}{\sigma} \qquad (H-11)$$

It is desired to generate a value for V (and by Eq. (H-11) for X) where V (and therefore X) is normally distributed such that

$$V = V(RN_i) \qquad (H-12)$$

that is, V is a normal random deviate. Given the RN_i as defined previously, and $S_n = \sum_{i=1}^{n} RN_i$, then by the Central Limit Theorem

$$V = (S_n - n\mu_r)/(n\sigma_r^2)^{1/2} \qquad (H-13)$$

is asymptotically normal. Because the distribution of the RN_i is uniform on the interval [0,1], $\mu_r = 1/2$ and $\sigma_r{}^2 = 1/12$. Consequently,

$$V = (S_n - n/2)(n/12)^{-1/2} \tag{H-14}$$

Asymptotic expansion with respect to n gives the cumulative distribution function of V:

$$F(v) = P(v) - \frac{Z(v)}{20n}[-v^3+3v] + \frac{Z(v)}{105n^2}[-v^5 + 10v^3 - 15v]+... \tag{H-15}$$

where Z(v) is the probability density function, or

$$Z(v) = e^{-v^2/2}/\sqrt{2\pi}$$

and

$$P(v) = \int_{-\infty}^{v} Z(u)du \tag{H-16}$$

Denote the error in the normal deviate V by $\Delta v \sim \Delta F(v)/Z(v)$. Then

$$\Delta v \sim [F(v) - P(v)]/Z(v)$$

$$\Delta v \sim [-4v^5 + (21n + 40)v^3 - (63n + 60)v]/420n^2 \tag{H-17}$$

For n = 12, the normal random deviate is

$$V = \sum_{i=1}^{12} RN_i - 6 \tag{H-18}$$

and the approximate error is

$$\Delta v \sim (-v^5 + 73v^3 - 204v)/15120 \tag{H-19}$$

For $|V| < 2$ the maximum error given by Eq. (H-19) is 0.01; for $2 < |V| < 3$, the maximum error given by Eq. (H-19) is 0.074. Thus, for 95% of the distribution, the maximum error in the normal deviate is 0.01. In the tail of the distribution, the error is greater and increases rapidly as $|V|$ increases.

Randomly generated values of X for the normal distribution may then be obtained by combining Eqs. (H-11) and (H-18); that is

$$x = \sigma v + \mu \tag{H-20}$$

If μ and σ are the mean and standard deviation of the normal distribution from which the lognormal distribution is derived by the variable change $U = e^X$, then randomly generated values of U are given by

$$u = e^{\sigma v + \mu}$$

for the lognormal distribution.

APPENDIX I - A PATHS Listing

```
      PROGRAM XPATH(INPUT=101B,OUTPUT=101B,TAPE5=101B,TAPE6=OUTPUT,TAPE2
     10=512,TAPE21=512,TAPE10=101B,DEBUG=OUTPUT,TAPE15=INPUT)
C
C           PATHS WITH PARTS OF SR PATHS COMBINED WITH XPATH
C           TO CONSOLIDATE ALL OF THE INPUT INTO XPATH.
C     INCLUDES ADDITIONAL PARTS OF PATHS 10/18/79
C
C
C
C     MFLAG INDICATES PATHFINDING OPTION CHOSEN
C       = .F. THEN MINIMUM TIME
C       = .T. THEN INTERRUPTION PROBABILITY
C
      LOGICAL MFLAG
      INTEGER DW,DAWT,DRT,PATH,Z,BEG,END,HEAD
      INTEGER YES,NO,TRUE,FALSE,NTRRPT,NSTEP,NLIN
      REAL NCNT
      DIMENSION NDNAM(130)
      DIMENSION NDPT(130)
      DIMENSION AC2(680), AC3(8), AC5(912)
      DIMENSION IBG(30), IND(30)
      COMMON W(130,3),PN(130),AWT(515,3),PA(515),RT(20,3),N,NA,NT
      COMMON /C1/ ITN(20),NTYP(130),IMPP(130),II(515),JJ(515),MR(515)
      COMMON /C2/ IW,IAWT,IRT,DW(130),DAWT(515),DRT(20),CPI,ITERM,RSPNS,
     1TIME
      COMMON /C3/ NODE(130,4),IREG(65,2),N12,N12P,IR,LI,IM,IR2,LI2,IM2
      COMMON /C4/ PLBL(130),ITEMP(130),IPERM(130),BIG
      COMMON /C5/ IRUN,NPTH,PATH(690),BEG(30),END(30),UM(30),UM2(30)
      COMMON /C6/ IMP(130)
      COMMON /C7/ MFLAG
      COMMON /C8/ HEAD(5),Z(130),PCNT(30),DUMP,MRUN,ISEED,PAI,DM,DM2
      COMMON /C9/ NCNT(130),CM(30),CM2(30),UMIN
      COMMON /C10/ PAI2
      EQUIVALENCE (AC2(1),IW)
      EQUIVALENCE (AC3(1),N12)
      EQUIVALENCE (AC5(1),IRUN)
C
      DATA YES/1HY/,NO/1HN/,TRUE/1HT/,FALSE/1HF/,NTRRPT/1HI/
      DATA NSTEP/1HS/,NLIN/1HL/
      DATA MAXNP/30/
      DATA IZRO/1H0/
      DATA IBNK/1H /
      DATA ONE/1.0/
C
      BIG=1.0E321
      REWIND 20
C
      CALL CLRY (W,414)
      CALL CLRY (PN,130)
      CALL CLRY (AWT,1545)
      CALL CLRY (PA,515)
      CALL CLRY (RT,60)
      CALL CLRY (ITN,20)
      CALL CLRY (NTYP,130)
      CALL CLRY (IMPP,130)
      CALL CLRY (II,515)
      CALL CLRY (JJ,515)
      CALL CLRY (MR,515)
      CALL CLRY (NODE,552)
      CALL CLRY (IREG,130)
      CALL CLRY (PLBL,130)
      CALL CLRY (ITEMP,130)
      CALL CLRY (IPERM,130)
      CALL CLRY (IMP,130)
C
      CALL CLRY (AC2,600)
      CALL CLRY (AC3,8)
      CALL CLRY (AC5,912)
      CALL CLRY (NDNAM,130)
C
      DO 10 I=1,5
      HEAD(I)=IBNK
   10 CONTINUE
C
C
      NA=0
      N=0
      NT=0
      MFLAG=.TRUE.
C
   20 WRITE (6,1080)
      WRITE (6,1090)
      READ (15,1120) IBRF
      IF (EOF(15).NE.0.0) GO TO 20
C
   30 WRITE (6,1100)
      WRITE (6,1110)
   40 READ (15,1120) IX
      IF (EOF(15).NE.0.0) GO TO 30
      IF (IX.EQ.TRUE) MFLAG=.FALSE.
      IF (IX.EQ.NTRAPT) MFLAG=.TRUE.
      IF (IX.EQ.TRUE.OR.IX.EQ.NTRAPT) GO TO 50
      WRITE (6,1130)
      GO TO 40
C
   50 READ (20) N1,N2,N3,N,NA,NR
      WRITE (6,1170) N1,N2,N3,NA
C
C     READ NODE WEIGHTS AND NODE DETECTION PROBABILITIES
      READ (20) (W(I,1),PN(I),I=1,N)
C
```

```
C     INITIALIZE TYPE OF DETECTION
      IF (.NOT.MFLAG) GO TO 110
   60 WRITE (6,1140)
      WRITE (6,1150)
      WRITE (6,1160)
      READ (15,1120) JTYP
      IF (EOF(15).NE.0.0) GO TO 60
      IF (JTYP.EQ.NLIN) GO TO 90
      IF (JTYP.EQ.NSTEP) GO TO 70
      WRITE (6,1130)
      GO TO 60
   70 DO 80 I=1,N
   80 NTYP(I)=2
      GO TO 110
   90 DO 100 I=1,N
  100 NTYP(I)=1
C
C     READ THE REGION, NODES CONNECTED AND ARC WEIGHT FOR EACH ARC.
C     COMPUTE ARC DETECTION PROBABILITIES.
  110 READ (20) (MR(IA),II(IA),JJ(IA),AWT(IA,1),IA=1,NA)
      IBEG=1
  120 LIKE=MR(IBEG)
      DO 130 I=IBEG,NA
      IF (MR(I).NE.LIKE) GO TO 140
  130 CONTINUE
      IEND=NA
      GO TO 150
  140 IEND=I-1
  150 XMAX=-9.9
      DO 160 I=IBEG,IEND
      IF (AWT(I,1).GT.XMAX) XMAX=AWT(I,1)
  160 CONTINUE
      DO 180 I=IBEG,IEND
      IF (AWT(I,1).GT.0.0) GO TO 170
      PA(I)=0.0
      GO TO 180
  170 PAT=0.0009/XMAX*AWT(I,1)+0.000105
      PA(I)=.00001*AINT(PAT*100000.0)
  180 CONTINUE
      IF (IEND.GE.NA) GO TO 190
      IBEG=IEND+1
      GO TO 120
  190 CONTINUE
C
C     PROMPT USER FOR TARGETS AND RESPONSE TIMES.
C     IF MINIMIZATION MEASURE IS TIME, DEFAULT RESPONSE
C     TIMES TO INFINITY.
C
      READ (20) (NDNAM(I),I=1,N)
      WRITE (6,1180) (NDNAM(I),I=1,N1)
  200 WRITE (6,1190) N1
      READ * , NT
      IF (EOF(15).NE.0.0) GO TO 200
      IF (NT.LE.N1) GO TO 210
      WRITE (6,1200)
      GO TO 200
C
  210 CONTINUE
      WRITE (6,1210)
      READ * , (ITN(IT),IT=1,NT)
      IF (EOF(15).NE.0.0) GO TO 210
      DO 240 I=1,NT
      KFLAG=0
      DO 220 IT=1,N
      IF (NDNAM(IT).NE.ITN(I)) GO TO 220
      ITN(I)=IT
      KFLAG=1
      GO TO 230
  220 CONTINUE
  230 IF (KFLAG.EQ.1) GO TO 240
      WRITE (6,1220) ITN(I)
      GO TO 210
  240 CONTINUE
C
      IF (MFLAG) GO TO 260
      DO 250 I=1,NT
  250 RT(I,1)=BIG
      GO TO 280
C
  260 CONTINUE
      DO 270 M=1,NT
      ITM=ITN(M)
      NDPT(M)=NDNAM(ITM)
  270 CONTINUE
C
      WRITE (6,1230) (NDPT(J),J=1,NT)
      IF (NT.GT.4) WRITE (6,1240)
      READ * , (RT(I,1),I=1,NT)
      IF (EOF(15).NE.0.0) GO TO 260
C
      WRITE (6,1250) (NDPT(IT),RT(IT,1),IT=1,NT)
      CALL PAWZ
C
  280 CONTINUE
C
C
      DUMP=0
      XDEV=0
      IF (IBRF.EQ.YES) GO TO 350
      WRITE (6,1260)
  290 CONTINUE
      WRITE (6,1270)
      READ * , ISEED
      IF (EOF(15).NE.0.0) GO TO 290
  300 CONTINUE
      WRITE (6,1280)
      WRITE (6,1290)
      READ * , MRUN
      FMRN=FLOAT(MRUN)
      IF (EOF(15).NE.0.0) GO TO 300
      IF (MFLAG) WRITE (6,1300)
      IF (.NOT.MFLAG) GO TO 310
      READ * , IRUN
      FIRN=FLOAT(IRUN)
  310 CONTINUE
C
  320 WRITE (6,1310)
      WRITE (6,1320)
      READ * , IW
      IF (EOF(15).NE.0.0) GO TO 320
  330 CONTINUE
      WRITE (6,1330)
      READ * , IAWT
      IF (EOF(15).NE.0.0) GO TO 330
      IF (MFLAG) WRITE (6,1340)
      IF (.NOT.MFLAG) GO TO 340
      READ * , IRT
  340 CONTINUE
      GO TO 380
C
  350 CONTINUE
      WRITE (6,1350)
      IF (MFLAG) WRITE (6,1360)
      IF (.NOT.MFLAG) GO TO 360
      READ * , ISEED,MRUN,IRUN,IW,IAWT,IRT
      FMRN=FLOAT(MRUN)
      FIRN=FLOAT(IRUN)
      GO TO 370
  360 CONTINUE
      IF (.NOT.MFLAG) WRITE (6,1370)
      READ * , ISEED,MRUN,IW,IAWT
      FMRN=FLOAT(MRUN)
  370 CONTINUE
C
  380 CALL RANSET (ISEED)
      IF (.NOT.MFLAG) IRT=1
      CPI=1.
      N1P=N1+1
      N12=N1+N2
      N12P=N12+1
      N=N12+N3
C
C
C     SET THE ARRAYS NODE AND IREG FOR USE IN FUNCTION IJAD
      DO 390 I=1,N
      NODE(I,1)=0
  390 NODE(I,3)=0
      L=1
  400 K=1
      IR=MR(L)
      IREG(IR,1)=L-1
      I=II(L)
      IF (NODE(I,1).EQ.0) GO TO 410
      NODE(I,3)=IR
      NODE(I,4)=K
      GO TO 420
  410 NODE(I,1)=IR
      NODE(I,2)=K
  420 K=K+1
      J=JJ(L)
      IF (NODE(J,1).EQ.0) GO TO 430
      NODE(J,3)=IR
      NODE(J,4)=K
      GO TO 440
  430 NODE(J,1)=IR
      NODE(J,2)=K
  440 IF (L.EQ.NA) GO TO 450
      L=L+1
      IF ((I.EQ.II(L)).AND.(IR.EQ.MR(L))) GO TO 420
      IREG(IR,2)=K
      L=L-K+K*(K-1)/2+1
      IF (L.LE.NA) GO TO 400
  450 IREG(IR,2)=K
C
C     SET DEFAULT PARAMETERS
      CALL DFLT
C
C     MODIFY OR EXAMINE INDIV ODE, OR RESPONSE TIME
C     DISTRIBUTIONS OR PARAMETERS.
C
      WRITE (6,1380)
      IF (IBRF.EQ.NO) WRITE (6,1390)
      IF (IBRF.EQ.NO) WRITE (6,1400)
  460 CONTINUE
      WRITE (6,1410)
      READ (15,1420) IVAR
      IF (EOF(15).NE.0.0) GO TO 460
      IF (IVAR.EQ.NO) GO TO 580
      IF (IBRF.EQ.YES) GO TO 470
      WRITE (6,1430)
```

```
C
  470 CONTINUE
      WRITE (6,1440)
      WRITE (6,1450)
      READ * , I
      IF (EOF(15).NE.0.0) GO TO 470
      IF (I.EQ.0) GO TO 580
      IF (.NOT.MFLAG.AND.I.EQ.5) GO TO 480
      GO TO (490,490,510,510,490), I
  480 WRITE (6,1460)
      GO TO 470
C
C     FIND NODE LABEL
C
  490 CONTINUE
      WRITE (6,1470)
      READ * , J
      IF (EOF(15).NE.0.0) GO TO 490
      CALL NDNUM (N,NDNAM,J)
      IF (J.EQ.0) GO TO 490
      IF (I.NE.5) GO TO 520
C
      DO 500 M=1,NT
      IF (ITN(M).NE.J) GO TO 500
      JT=M
      GO TO 520
  500 CONTINUE
      WRITE (6,1480) NDNAM(J)
      GO TO 490
C
C     FIND ARC ENDPOINTS
  510 CONTINUE
      WRITE (6,1490)
      READ * , J1,J2
      IF (EOF(15).NE.0.0) GO TO 510
      CALL NDNUM (N,NDNAM,J1)
      CALL NDNUM (N,NDNAM,J2)
      IF (J1*J2.EQ.0) GO TO 510
      IR=NODE(J1,1)
      LI=NODE(J1,2)
      IM=IREG(IR,1)+(LI-1)*IREG(IR,2)-LI*(LI+1)/2
      IR2=NODE(J1,3)
      LI2=NODE(J1,4)
      IF (IR2.NE.0) IM2=IREG(IR2,1)+(LI2-1)*IREG(IR2,2)-LI2*(LI2+1)/2
      J=IJAD(J2)
      IF (J.GT.0) GO TO 520
      WRITE (6,1500) NDNAM(J1),NDNAM(J2)
      GO TO 510
C
  520 GO TO (530,540,550,560,570), I
```

```
C                                                                       A3510
C     NODE TRAVERSAL TIME                                               A3520
C                                                                       A3530
  530 CONTINUE                                                          A3540
      WRITE (6,1510) NDNAM(J)                                           A3550
      CALL MOD (I,W(J,1),W(J,2),W(J,3),DW(J))                           A3560
      GO TO 470                                                         A3570
C                                                                       A3580
C     NODE DETECTION PROBABILITY                                        A3590
C                                                                       A3600
  540 CONTINUE                                                          A3610
      WRITE (6,1520) NDNAM(J)                                           A3620
      CALL MOD (I,PN(J),0,0,1)                                          A3630
      GO TO 470                                                         A3640
C                                                                       A3650
C     ARC TRAVERSAL TIME                                                A3660
C                                                                       A3670
  550 CONTINUE                                                          A3680
      WRITE (6,1530) NDNAM(J1),NDNAM(J2)                                A3690
      CALL MOD (I,AWT(J,1),AWT(J,2),AWT(J,3),DAWT(J))                   A3700
      GO TO 470                                                         A3710
C                                                                       A3720
C     ARC DETECTION PROBABILITY                                         A3730
C                                                                       A3740
  560 CONTINUE                                                          A3750
      WRITE (6,1540) NDNAM(J1),NDNAM(J2)                                A3760
      CALL MOD (I,PA(J),0,0,1)                                          A3770
      GO TO 470                                                         A3780
C                                                                       A3790
C     RESPONSE TIME FOR TERMINAL NODE                                   A3800
C                                                                       A3810
  570 CONTINUE                                                          A3820
      WRITE (6,1550) NDNAM(J)                                           A3830
      CALL MOD (I,RT(JT,1),RT(JT,2),RT(JT,3),DRT(JT))                   A3840
      GO TO 470                                                         A3850
C                                                                       A3860
C                                                                       A3870
C     END MODIFICATONS                                                  A3880
  580 CONTINUE                                                          A3890
      WRITE (6,1560)                                                    A3900
      WRITE (6,1570)                                                    A3910
      READ (15,1580) (HEAD(I),I=1,5)                                    A3920
      IF (EOF(15).NE.0.0) GO TO 586                                     A3930
C                                                                       A3940
      CALL PATHS                                                        A3950
C                                                                       A3960
C                                                                       A3970
C     RUN APPROPRIATE PHASE 2 SUBROUTINE DEPENDING ON                   A3980
C     MINIMIZATION MEASURE.                                             A3990
      IF (IRUN.GT.0) CALL EZ                                            A4000
      IF (.NOT.MFLAG) CALL THYME (UM,UM2,W,AWT,DW,DAWT,BEG,END,PATH,NPTH A4010
     1)                                                                  A4020
C                                                                        A4030
C     COMPUTE NODE CRITICALITY COUNTS AND DETERMINE                      A4040
C     THE MINIMUM UNCONDITIONAL (PHASE 2) MEASURE.                       A4050
      UMIN=BIG                                                           A4060
      DO 610 NP=1,NPTH                                                   A4070
      IF (UM(NP).GE.UMIN) GO TO 590                                      A4080
      UMIN=UM(NP)                                                        A4090
      KP=NP                                                              A4100
  590 IBEG=BEG(NP)                                                       A4110
      IEND=END(NP)                                                       A4120
      DO 600 J=IBEG,IEND                                                 A4130
      NOD=PATH(J)                                                        A4140
  600 NCNT(NOD)=NCNT(NOD)+PCNT(NP)                                       A4150
  610 CONTINUE                                                           A4160
      IF (IRUN.GT.0) UMIN=UMIN/FIRN                                      A4170
      CALL PAWZ                                                          A4180
C     INTERRUPTION MEASURE INFORMATION                                   A4190
C                                                                        A4200
C                                                                        A4210
C     SUMMARY INFORMATION                                                A4220
      WRITE (6,1590)                                                     A4230
      WRITE (6,1300)                                                     A4240
      WRITE (6,1600)                                                     A4250
      WRITE (6,1610)                                                     A4260
      IF (HEAD(1).NE.IBNK) WRITE (6,1620) (HEAD(I),I=1,5)                A4270
      IF (HEAD(1).NE.IBNK) WRITE (6,1610)                                A4280
      WRITE (6,1630) ISEED                                               A4290
      WRITE (6,1610)                                                     A4300
      WRITE (6,1640) NPTH                                                A4310
      WRITE (6,1610)                                                     A4320
      DMR=DUMP/FMRN                                                      A4330
      WRITE (6,1650) DMR                                                 A4340
      WRITE (6,1610)                                                     A4350
      WRITE (6,1660) MRUN                                                A4360
      WRITE (6,1610)                                                     A4370
      IF (MFLAG) GO TO 620                                               A4380
C                                                                        A4390
C     TIME MEASURE INFORMATION                                           A4400
C                                                                        A4410
      WRITE (6,1670)                                                     A4420
      WRITE (6,1610)                                                     A4430
      WRITE (6,1680) UMIN                                                A4440
      WRITE (6,1610)                                                     A4450
      WRITE (6,1690) UM2(KP)                                             A4460
      WRITE (6,1610)                                                     A4470
      RLU=PAI/(FMRN*UMIN)                                                A4480
      WRITE (6,1700) RLU                                                 A4490
      WRITE (6,1610)                                                     A4500
      GO TO 680
C
  620 CONTINUE
      WRITE (6,1710) IRUN
      WRITE (6,1610)
      WRITE (6,1720)
      WRITE (6,1610)
      VPAI=PAI2-PAI**2/FMRN
      IF (VPAI.LT.0.) VPAI=0.
      IF (MRUN.EQ.1) GO TO 630
      DEV=SQRT(VPAI/(FMRN-1.))
  630 CONTINUE
      WRITE (6,1730)
      WRITE (6,1740)
      FMN=PAI/FMRN
      STDVM=DEV/SQRT(FMRN)
C
C     FMN CAN BE ZERO IF ALL PROBABILITY OF INTERRUPTION VALUES FOR ALL
C         CRITICAL PATHS ARE ZERO.
C
      IF (FMN.EQ.0.) GO TO 640
      COV=DEV/FMN
      WRITE (6,1750) FMN,DEV,STDVM,COV
      GO TO 650
  640 WRITE (6,1760) FMN,DEV,STDVM
  650 IF (IRUN.EQ.0) GO TO 680
      WRITE (6,1610)
      VUMKP=UM2(KP)-UM(KP)**2/FIRN
      IF (VUMKP.LT.0.) VUMKP=0.
      UDEV=SQRT(VUMKP)
      IF (IRUN.GT.1) UDEV=UDEV/SQRT((FIRN-1.))
      UDVM=UDEV/SQRT(FIRN/1.)
C
C     UMIN CAN BE ZERO IF THE MEAN PROBABILITY OF INTERRUPTION
C          CALCULATED IN PHASE 2 WAS ZERO FOR SOME PATH.
C
      IF (UMIN.EQ.0.) GO TO 660
      UCOV=UDEV/UMIN
      WRITE (6,1770) UMIN,UDEV,UDVM,UCOV
      GO TO 670
  660 WRITE (7,1780) UMIN,UDEV,UDVM
  670 WRITE (6,1610)
  680 CONTINUE
      WRITE (6,1600)
      CALL PAWZ
C
C     PATH DESCRIPTIONS
C
  690 CONTINUE
      WRITE (6,1790)
      READ (15,1420) JVAR
      IF (EOF(15).NE.0.0) GO TO 690
      CALL SORT (PCNT,Z,1,NPTH,KPTH,BIG,0)
      IF (JVAR.EQ.MO) GO TO 1000
      WRITE (6,1010)
      WRITE (6,1300)
      WRITE (6,1020)
      WRITE (6,1060)
      WRITE (6,1030)
      WRITE (6,1060)
      DO 720 NP=1,KPTH
      K=Z(NP)
      IBEG=BEG(K)
      IEND=END(K)
      L=0
      DO 710 M=IBEG,IEND
      IPT=PATH(M)
      NDN=NDNAM(IPT)
      IF (NDN.GE.1000) GO TO 700
      L=L+1
      NDPT(L)=NDN
  700 CONTINUE
  710 CONTINUE
      IF (L+1.NE.10) WRITE (6,1040) NP,(NDPT(J),J=1,L)
      IF (L+1.EQ.10) WRITE (6,1050) NP,(NDPT(J),J=1,L)
      WRITE (6,1060)
  720 CONTINUE
      WRITE (6,1020)
C
C     MODIFY DUMP IF ANY SAVED PATHS ARE DELETED
      IF (KPTH.EQ.NPTH) GO TO 740
      KP1=KPTH+1
      DO 730 NP=KP1,NPTH
      DUMP=DUMP+PCNT(NP)
      K=Z(NP)
      DM=DM+CM(K)
      DM2=DM2+CM2(K)
  730 CONTINUE
  740 CALL PAWZ
  750 CONTINUE
      WRITE (6,1000)
      READ (15,1420) IVAR
      IF (EOF(15).NE.0.0) GO TO 750
      IF (MFLAG) GO TO 870
C
C     RANKING/TIME INFORMATION
C
      WRITE (6,1070)
      WRITE (6,1080)
      WRITE (6,1090)
      WRITE (6,1900)
      IF (IVAR.EQ.NO) GO TO 760
      WRITE (6,1910)
      WRITE (6,1930)
      GO TO 770
  760 WRITE (6,1920)
      WRITE (6,1940)
  770 CUM=0
      COR=0
      UMS=0
      UMS2=0
      CXU=0
      CRT2=0
      DO 830 NP=1,KPTH
      K=Z(NP)
      CNT=PCNT(NP)
      CRIT=CNT/FMRN
      CUM=CUM+CRIT
C
      CDEV=0.0
      IF (CNT.EQ.1.0) GO TO 780
      CDEV2=(CM2(K)-CM(K)**2/CNT)/(CNT-1.0)
      IF (CDEV2.GT.0.0) CDEV=SQRT(CDEV2)
  780 CONTINUE
      WRITE (6,2000)
      CMC=CM(K)/CNT
      IF (CNT.LT.MRUN) GO TO 790
      XMIN=0.0
      GO TO 800
  790 XMIN=(UM(K)-CRIT*CMC)/(1.-CRIT)
      IF (XMIN.LT.0.0) XMIN=0.0
  800 IF (IVAR.EQ.NO) GO TO 810
      WRITE (6,1950) NP,CRIT,CUM,CMC,CDEV,UM(K),UM2(K),XMIN
      GO TO 820
  810 WRITE (6,1960) NP,CRIT,CUM,UM(K),UM2(K)
C     COMPUTE PARAMETERS FOR LINEAR CORRELATION COEFF
  820 UMS=UMS+UM(K)
      UMS2=UMS2+UM(K)**2
      CXU=CXU+UM(K)*CRIT
      CRT2=CRT2+CRIT**2
  830 CONTINUE
C     COMPUTE LINEAR CORRELATION COEFFICIENT
      IF (UMS.EQ.0.0) GO TO 840
      COR=(UMS*CUM-NPTH*CXU)/SQRT((NPTH*CRT2-1.)*(NPTH*UMS2-UMS**2))
      WRITE (6,2000)
      WRITE (6,1970) COR
      GO TO 850
  840 COR=0.0
      WRITE (6,2000)
      WRITE (6,1970) COR
  850 CONTINUE
      WRITE (6,2000)
      IF (DUMP.EQ.0.0) GO TO 860
      CRIT=DUMP/FMRN
      CUM=CUM+CRIT
      VDM=DM2-DM**2/DUMP
      IF (VDM.LT.0.) VDM=0.
      XDEV=SQRT(VDM)
      IF (DUMP.GT.1.) XDEV=XDEV/SQRT(DUMP-1.)
      DMP=DM/DUMP
      WRITE (6,1980) CRIT,CUM,DMP,XDEV
      WRITE (6,2000)
  860 CONTINUE
      WRITE (6,1990)
      CALL PAWZ
      GO TO 1000
C
C     PATH RANKING/INTERRUPTION INFORMATION
C
  870 CONTINUE
      WRITE (6,2010)
      WRITE (6,2020)
      WRITE (6,2030)
      WRITE (6,2040)
      IF (IVAR.EQ.NO) GO TO 880
      WRITE (6,2050)
      WRITE (6,2070)
      GO TO 890
  880 WRITE (6,2060)
      WRITE (6,2080)
C
  890 CUM=0
      COR=0
      UMS=0
      UMS2=0
      CXU=0
      CRT2=0
      DO 960 NP=1,KPTH
      K=Z(NP)
      CNT=PCNT(NP)
      CRIT=CNT/FMRN
      CUM=CUM+CRIT
C
      CDEV=0.0
      DEV=0.0
      IF (CNT.EQ.1.0) GO TO 900
      CDEV2=(CM2(K)-CM(K)**2/CNT)/(CNT-1.0)
      IF (CDEV2.GT.0.0) CDEV=SQRT(CDEV2)
  900 IF (IRUN.LE.1) GO TO 910
      DEV2=(UM2(K)-UM(K)**2/FIRN)/(FIRN-1.0)
      IF (DEV2.GT.0.0) DEV=SQRT(DEV2)                                   A6510
  910 CONTINUE                                                          A6520
      WRITE (6,2030)                                                    A6530
      CMC=CM(K)/CNT                                                     A6540
      IF (IRUN.EQ.0) GO TO 930                                          A6550
      UMR=UM(K)/FIRN                                                    A6560
      IF (CNT.LT.MRUN) GO TO 920                                        A6570
      XMIN=0.0                                                          A6580
      GO TO 930                                                         A6590
  920 XMIN=(UMR-CRIT*CMC)/(1.-CRIT)                                     A6600
      IF (XMIN.LT.0.0) XMIN=0.0                                         A6610
      IF (XMIN.GT.1.0) XMIN=1.0                                         A6620
  930 IF (IVAR.EQ.NO) GO TO 940                                         A6630
      IF (IRUN.GT.0) WRITE (6,2090) NP,CRIT,CUM,CMC,CDEV,UMR,DEV,XMIN   A6640
      IF (IRUN.EQ.0) WRITE (6,2110) NP,CRIT,CUM,CMC,CDEV                A6650
      GO TO 950                                                         A6660
  940 IF (IRUN.GT.0) WRITE (6,2100) NP,CRIT,CUM,UMR,DEV                 A6670
      IF (IRUN.EQ.0) WRITE (6,2120) NP,CRIT,CUM                         A6680
C     COMPUTE PARAMETERS FOR LINEAR CORRELATION COEFF                   A6690
  950 IF (IRUN.EQ.0) GO TO 960                                          A6700
      UMS=UMS+UMR                                                       A6710
      UMS2=UMS2+UMR**2                                                  A6720
      CXU=CXU+UMR*CRIT                                                  A6730
      CRT2=CRT2+CRIT**2                                                 A6740
  960 CONTINUE                                                          A6750
C     COMPUTE LINEAR CORRELATION COEFFICIENT                            A6760
      IF (IRUN.EQ.0) GO TO 980                                          A6770
      IF (UMS.EQ.0.0) GO TO 970                                         A6780
      COR=(UMS*CUM-NPTH*CXU)/SQRT((NPTH*CRT2-1.)*(NPTH*UMS2-UMS**2))    A6790
      WRITE (6,2030)                                                    A6800
      WRITE (6,1970) COR                                                A6810
      GO TO 980                                                         A6820
  970 COR=0.0                                                           A6830
      WRITE (6,2030)                                                    A6840
      WRITE (6,1970) COR                                                A6850
  980 CONTINUE                                                          A6860
C                                                                       A6870
C     IF THE DUMP IS NOT EMPTY DISPLAY IT                               A6880
      WRITE (6,2030)                                                    A6890
      IF (DUMP.EQ.0.0) GO TO 990                                        A6900
      CRIT=DUMP/FMRN                                                    A6910
      CUM=CUM+CRIT                                                      A6920
      VDM=DM2-DM**2/DUMP                                                A6930
      IF (VDM.LT.0.) VDM=0.                                             A6940
      XDEV=SQRT(VDM)                                                    A6950
      IF (DUMP.GT.1.) XDEV=XDEV/SQRT(DUMP-1.)                           A6960
      DMP=DM/DUMP                                                       A6970
      WRITE (6,2130) CRIT,CUM,DMP,XDEV                                  A6980
      WRITE (6,2030)                                                    A6990
  990 CONTINUE                                                          A7000
```

```
      WRITE (6,2020)
      CALL PAWZ
C
C     STORE INFORMATION ON TAPES 21 AND 5 FOR USE AFTER
C     PROGRAM EXECUTION.
C
 1000 WRITE (21) (PA(I),I=1,NA)
      WRITE (21) KPTH
C
      DO 1010 I=1,KPTH
      IZL=Z(I)
      IBG(I)=BEG(IZL)
      IND(I)=END(IZL)
 1010 CONTINUE
C
      WRITE (21) (IBG(I),I=1,KPTH),(IND(I),I=1,KPTH)
      L=END(NPTH)
      WRITE (21) L,(PATH(I),I=1,L)
C
C     EXTRA INFO ON PI AND LOWER BOUND
C
      IF (.NOT.MFLAG) GO TO 1020
      WRITE (21) ONE
      PFM=PAI/FMRN
      WRITE (21) PFM
      GO TO 1030
C
 1020 WRITE (21) IZRO
 1030 DO 1040 NP=1,KPTH
      K=Z(NP)
      UMR=UM(K)/FIRN
      IF (MFLAG) WRITE (21) NP,UMR
      IF (.NOT.MFLAG) WRITE (21) NP,UM(K)
 1040 CONTINUE
C
      WRITE (5,2140) KPTH,IRUN
      IF (IRUN.EQ.0) GO TO 1060
      DO 1050 NP=1,KPTH
      K=Z(NP)
      PCR=PCNT(NP)/FMRN
      CMP=CM(K)/PCNT(NP)
      UMR=UM(K)/FIRN
      WRITE (5,2150) PCR,CMP,UMR
 1050 CONTINUE
 1060 CONTINUE
      IF (JVAR.EQ.NO) GO TO 1070
C
C     NODE CRITICALITY INFORMATION
C
      FMRN5=.005*FMRN
      CALL SORT (NCNT,Z,1,N1,K1,BIG,FMRN5)
      CALL SORT (NCNT,Z,N1P,N12,K12,BIG,FMRN5)
      CALL SORT (NCNT,Z,N12P,N,K,BIG,FMRN5)
C
      WRITE (6,2160)
      WRITE (6,2170)
      WRITE (6,2180)
      WRITE (6,2190)
      CALL PRNT (Z,NCNT,FMRN,1,N1,K1,NDNAM)
C
      WRITE (6,1300)
      WRITE (6,2200)
      WRITE (6,2190)
      CALL PRNT (Z,NCNT,FMRN,N12P,N,K,NDNAM)
C
      WRITE (6,2210)
      WRITE (6,2220)
      WRITE (6,2190)
      CALL PRNT (Z,NCNT,FMRN,N1P,N12,K12,NDNAM)
      WRITE (6,2170)
      CALL PAWZ
 1070 CONTINUE
      STOP
C
C
C
 1080 FORMAT (22X,11HPATHS INPUT//5X,14HENTER: Y = YES/12X,6HN = NO)
 1090 FORMAT (29HDO YOU WANT BRIEF INPUT MODE )
 1100 FORMAT (/5X,15HENTER: T = TIME/12X,28HI = INTERRUPTION PROBABILITY
     1)
 1110 FORMAT (41HMINIMIZATION MEASURE TO BE USED BY PATHS )
 1120 FORMAT (A1)
 1130 FORMAT (/24HINVALID INPUT. RE-ENTER.)
 1140 FORMAT (/5X,46HENTER: L = LINEAR FUNCTION OF NODE PENETRATION)
 1150 FORMAT (12X,44HS = STEP: FULL. VALUE AT COMPLETE PENETRATION)
 1160 FORMAT (51HSELECT NODE DETECTION TYPE: LINEAR OR STEP FUNCTION)
 1170 FORMAT (/I3,13H TARGET NODES,I5,14H BARRIER NODES,I5,15H BOUNDARY
     1NODES,I5,5H ARCS/)
 1180 FORMAT (/27H LIST OF ALL TARGET NODES: ,/(12I5))
 1190 FORMAT (/49H ENTER: NUMBER OF TERMINALS (TARGETS AND/OR OTHER,32H
     1FACILITY NODES) TO BE EXAMINED./39H       (NOTE: MAXIMUM NO. OF TE
     2RMINALS=,I3,1H),)
 1200 FORMAT (37HMAXIMUM NUMBER OF TERMINALS EXCEEDED.,31H  RE-ENTER NUM
     1BER OF TERMINALS.)
 1210 FORMAT (42HENTER: LABELS OF TERMINALS TO BE EXAMINED )
 1220 FORMAT (18HINVALID NODE LABEL,I5,19H ENTERED.  RE-ENTER,26H ALL TE
     1RMINAL NODE LABELS.)
 1230 FORMAT (/42H ENTER: RESPONSE TIMES FOR TARGETS LABELED,4I5,/,(12I5
     1))
 1240 FORMAT (1H )
```

```
1250 FORMAT (/24H TARGET    RESPONSE TIME/,(I5,2X,F13.2))                A8010
1260 FORMAT (1H )                                                        A8020
1270 FORMAT (44HENTER: SEED FOR THE RANDOM NUMBER GENERATOR )            A8030
1280 FORMAT (1H )                                                        A8040
1290 FORMAT (50HENTER: NUMBER OF PATHFINDING REPLICATIONS DESIRED )      A8050
1300 FORMAT (41HENTER: NUMBER OF INTERRUPTION CALCULATION,25H REPLICATI  A8060
     1ONS/PATH SAVED )                                                   A8070
1310 FORMAT (/5X,17HDISTRIBUTION CODE//5X,8H0 = NULL/5X,17H1 = DETERMIN  A8080
     1ISTIC/5X,11H2 = UNIFORM/5X,14H3 = TRIANGULAR/5X,10H4 = NORMAL/5X,1 A8090
     23H5 = LOGNORMAL/)                                                  A8100
1320 FORMAT (50HENTER: DISTRIBUTION CODE FOR NODE TRAVERSAL TIMES )      A8110
1330 FORMAT (49HENTER: DISTRIBUTION CODE FOR ARC TRAVERSAL TIMES )       A8120
1340 FORMAT (44HENTER: DISTRIBUTION CODE FOR RESPONSE TIMES )            A8130
1350 FORMAT (1H )                                                        A8140
1360 FORMAT (35HENTER: SEED,MRUN,IRUN, DW,DAWT,DRT )                     A8150
1370 FORMAT (25HENTER: SEED,MRUN,DW,DAWT )                               A8160
1380 FORMAT (1H )                                                        A8170
1390 FORMAT (20H     ENTER - Y = YES)                                    A8180
1400 FORMAT (19H             N = NO)                                     A8190
1410 FORMAT (51HDO YOU WANT TO MODIFY OR EXAMINE ANY DISTRIBUTIONS )     A8200
1420 FORMAT (A1)                                                         A8210
1430 FORMAT (/14X,9HTYPE CODE//5X,27H0 = TERMINATE MODIFICATIONS/5X,23H  A8220
     11 = NODE TRAVERSAL TIME/5X,30H2 = NODE DETECTION PROBABILITY/5X,22 A8230
     2H3 = ARC TRAVERSAL TIME/5X,29H4 = ARC DETECTION PROBABILITY/5X,31H A8240
     35 = TERMINAL NODE RESPONSE TIME//)                                 A8250
1440 FORMAT (1H )                                                        A8260
1450 FORMAT (17HENTER: TYPE CODE )                                       A8270
1460 FORMAT (28HCANNOT MODIFY RESPONSE TIMES)                            A8280
1470 FORMAT (18HENTER: NODE LABEL )                                      A8290
1480 FORMAT (I11,29H IS NOT A TERMINAL NODE LABEL)                       A8300
1490 FORMAT (23HENTER: NODE LABEL PAIR )                                 A8310
1500 FORMAT (11X,1H(,I4,1H,,I4,16H) IS NOT AN ARC )                      A8320
1510 FORMAT (/8X,4HNODE,I5,15H TRAVERSAL TIME)                           A8330
1520 FORMAT (/8X,4HNODE,I5,22H DETECTION PROBABILITY)                    A8340
1530 FORMAT (/8X,5HARC (,I4,1H,,I4,16H) TRAVERSAL TIME)                  A8350
1540 FORMAT (/8X,5HARC (,I4,1H,,I4,23H) DETECTION PROBABILITY)           A8360
1550 FORMAT (/8X,31HRESPONSE TIME FOR TERMINAL NODE,I5)                  A8370
1560 FORMAT (1H )                                                        A8380
1570 FORMAT (14HENTER: HEADER )                                          A8390
1580 FORMAT (6A10)                                                       A8400
1590 FORMAT (36H                      PATHS SUMMARY )                    A8410
1600 FORMAT (2X,30(2H* ))                                                A8420
1610 FORMAT (2H *,59X,1H*)                                               A8430
1620 FORMAT (6X,5A10)                                                    A8440
1630 FORMAT (36H      RANDOM NUMBER GENERATOR SEED =,I4)                 A8450
1640 FORMAT (38H      NUMBER OF DISTINCT PATHS SAVED =,I3)               A8460
1650 FORMAT (48H      PERCENTAGE OF PATHS FOUND BUT NOT SAVED  =,F5.3)   A8470
1660 FORMAT (42H      NUMBER OF PATHFINDING REPLICATIONS =,I4)           A8480
1670 FORMAT (44H     UNCONDITIONAL MEAN PATH TRAVERSAL TIME )            A8490
1680 FORMAT (20H           MINIMUM =,F7.2)                               A8500
```

```
1690 FORMAT (20H                 STDV  =,F7.2)                           A8510
1700 FORMAT (38H      (RATIO OF LOWER TO UPPER BOUND = ,F4.2,2H ))       A8520
1710 FORMAT (54H        NUMBER OF INTERRUPTION REPLICATIONS/PATH SAVED =, A8530
  1I4)                                                                   A8540
1720 FORMAT (43H        PROBABILITY OF ADVERSARY INTERRUPTION)           A8550
1730 FORMAT (10X,8H  MEAN  ,11H  STD DEV  ,11H  STD DEV  ,7H  COV  )     A8560
1740 FORMAT (2H *,37X,7HOF MEAN,15X,1H*)                                 A8570
1750 FORMAT (5X,11HLOWER BOUND,4X,F4.2,6X,F4.2,7X,F4.2,4X,F4.2)          A8580
1760 FORMAT (5X,11HLOWER BOUND,4X,F4.2,6X,F4.2,7X,F4.2,4X,9HUNDEFINED)   A8590
1770 FORMAT (5X,7HMINIMUM,8X,F4.2,6X,F4.2,7X,F4.2,4X,F4.2)               A8600
1780 FORMAT (5X,7HMINIMUM,8X,F4.2,6X,F4.2,7X,F4.2,4X,9HUNDEFINED)        A8610
1790 FORMAT (33H DO YOU WANT TO SEE PATHS OUTPUT )                       A8620
1800 FORMAT (49HDO YOU WANT EXTENDED RATHER THAN STANDARD OUTPUT )       A8630
1810 FORMAT (40H                             PATH DESCRIPTION,/)         A8640
1820 FORMAT (2X,34(2H* ))                                                A8650
1830 FORMAT (36H     PATH #              NODAL SEQUENCE)                 A8660
1840 FORMAT (I7,3X,11I5,/,(10X,11I5))                                    A8670
1850 FORMAT (I7,3X,11I5)                                                 A8680
1860 FORMAT (2H *,67X,1H*)                                               A8690
1870 FORMAT (/44H                  PATH RANKING/TIME INFORMATION/)       A8700
1880 FORMAT (2X,34(2H* ))                                                A8710
1890 FORMAT (2H *,67X,1H*)                                               A8720
1900 FORMAT (47H  PATH    RANKING           PATH TRAVERSAL TIMES )       A8730
1910 FORMAT (44H * NMBR    INDEX CUM         CONDITIONAL      ,26HUNCONDI A8740
  1TIONAL     NOT MIN *)                                                 A8750
1920 FORMAT (43H * NMBR    INDEX CUM         UNCONDITIONAL  ,26X,1H*)    A8760
1930 FORMAT (29X,38HMEAN   STDV      MEAN   STDV       MEAN)             A8770
1940 FORMAT (29X,11HMEAN   STDV)                                         A8780
1950 FORMAT (I7,4X,F4.2,2X,F4.2,4X,F8.2,F7.2,2X,F8.2,F7.2,2X,F8.2)       A8790
1960 FORMAT (I7,4X,F4.2,2X,F4.2,4X,F8.2,F7.2)                            A8800
1970 FORMAT (36H     LINEAR CORRELATION COEFFICIENT  ,F6.2)              A8810
1980 FORMAT (7X,1HD,5X,F4.2,2X,F4.2,8X,F4.2,1X,F4.2)                     A8820
1990 FORMAT (2X,34(2H* ))                                                A8830
2000 FORMAT (2H *,67X,1H*)                                               A8840
2010 FORMAT (/41H     PATH RANKING/INTERRUPTION INFORMATION/)            A8850
2020 FORMAT (2X,34(2H* ))                                                A8860
2030 FORMAT (2H *,67X,1H*)                                               A8870
2040 FORMAT (53H      PATH       RANKING       INTERRUPTION PROBABILITY ) A8880
2050 FORMAT (44H *  NUMBER    INDEX CUM      CONDITIONAL     ,26HUNCONDI A8890
  1TIONAL     NOT MIN *)                                                 A8900
2060 FORMAT (44H *  NUMBER    INDEX CUM      UNCONDITIONAL  ,25X,1H*)    A8910
2070 FORMAT (29X,38HMEAN   STDV      MEAN   STDV       MEAN)             A8920
2080 FORMAT (29X,11HMEAN   STDV)                                         A8930
2090 FORMAT (I8,5X,F4.2,2X,F4.2,6X,F4.2,3X,F4.2,6X,F4.2,3X,F4.2,6X,F4.2 A8940
  1)                                                                     A8950
2100 FORMAT (I8,5X,F4.2,2X,F4.2,6X,F4.2,3X,F4.2)                         A8960
2110 FORMAT (I8,5X,F4.2,2X,F4.2,6X,F4.2,3X,F4.2)                         A8970
2120 FORMAT (I8,5X,F4.2,2X,F4.2)                                         A8980
2130 FORMAT (7X,1HD,5X,F4.2,2X,F4.2,8X,F4.2,1X,F4.2)                     A8990
2140 FORMAT (2I6)                                                        A9000
2150 FORMAT (3F15.13)                                                    A9010
2160 FORMAT (/15X,24HNODE RANKING INFORMATION)                           A9020
2170 FORMAT (/2X,26(2H* )/)                                              A9030
2180 FORMAT (11H *   TARGET,4X,7HRANKING,9X,6HTARGET,4X,7HRANKING,5X,1H A9040
  1*)                                                                    A9050
2190 FORMAT (47H        NODE       INDEX            NODE      INDEX)     A9060
2200 FORMAT (54H *  BOUNDARY   RANKING         BOUNDARY   RANKING      *) A9070
2210 FORMAT (1H )                                                        A9080
2220 FORMAT (54H *  BARRIER    RANKING         BARRIER    RANKING      *) A9090
     END                                                                 A9100
```

```
      SUBROUTINE CHKPTH (ITERM)
C
C
      INTEGER ITERM,N1,N2,N3,IRG1,IRG2
      COMMON /C6/ IMP(130)
      N1=ITERM
      N2=IMP(N1)
      IRG1=IRG(N1,N2)
   10 N3=IMP(N2)
      IF (N3.EQ.0) GO TO 30
      IRG2=IRG(N2,N3)
      IF (IRG1.NE.IRG2) GO TO 20
      IMP(N1)=N3
      N2=N3
      GO TO 10
   20 CONTINUE
      N1=N2
      N2=N3
      IRG1=IRG2
      GO TO 10
   30 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CLRY (A,N)
C
C
C        CLEARS ARRAY A OF LENGTH N TO ZERO.
C
      DIMENSION A(N)
C
C
      DO 10 I=1,N
      A(I)=0
   10 CONTINUE
      RETURN
      END
      SUBROUTINE DFLT
C
C
C     SET DEFAULT PARAMETERS FOR TIMES ACCORDING TO THE DISTRIBUTIONS
C     SPECIFIED BY IW (NODES), IAWT (ARCS), AND IRT (RESPONSE TIMES).
C     WHEN MODE = 0 DISTRIBUTION IS AUTOMATICALLY DEFAULTED TO
C     DETERMINISTIC. SEE FUNCTION SAMPL FOR PARAMETER DESCRIPTIONS.
C     NOTE: PARAMETERS FOR NORMAL, LOGNORMAL, AND TRIANGULAR ARE
C     CURRENTLY SET TO PRODUCE A COEFFICIENT OF VARIATION OF .20
      INTEGER DW,DAWT,DRT
      COMMON W(130,3),PN(130),AWT(515,3),PA(515),RT(20,3),N,NA,NT
      COMMON /C2/ IW,IAWT,IRT,DW(130),DAWT(515),DRT(20),CPI,ITERM,RSPNS,
     1TIME
C
      DO 10 I=1,N
   10 DW(I)=1
      DO 20 I=1,NA
   20 DAWT(I)=1
      DO 30 I=1,NT
   30 DRT(I)=1
C
      GO TO (100,40,40,60,80), IW
C
   40 DO 50 I=1,N
      IF (W(I,1).EQ.0.) GO TO 50
      DW(I)=IW
      W(I,2)=.500*W(I,1)
      W(I,3)=1.5*W(I,1)
   50 CONTINUE
      GO TO 100
C
   60 DO 70 I=1,N
      IF (W(I,1).EQ.0.) GO TO 70
      DW(I)=IW
      W(I,2)=0.
      W(I,3)=.200*W(I,1)
   70 CONTINUE
      GO TO 100
C
   80 DO 90 I=1,N
      IF (W(I,1).EQ.0.) GO TO 90
      DW(I)=IW
      W(I,1)=ALOG(W(I,1))+.039
      W(I,3)=.20
   90 CONTINUE
C
  100 GO TO (170,110,110,130,150), IAWT
C
  110 DO 120 I=1,NA
      IF (AWT(I,1).EQ.0.) GO TO 120
      DAWT(I)=IAWT                                                      D 510
      AWT(I,2)=.500*AWT(I,1)                                            D 520
      AWT(I,3)=1.5*AWT(I,1)                                             D 530
  120 CONTINUE                                                          D 540
      GO TO 170                                                         D 550
C                                                                       D 560
  130 DO 140 I=1,NA                                                     D 570
      IF (AWT(I,1).EQ.0.) GO TO 140                                     D 580
      DAWT(I)=IAWT                                                      D 590
      AWT(I,2)=0.                                                       D 600
      AWT(I,3)=.200*AWT(I,1)                                            D 610
  140 CONTINUE                                                          D 620
      GO TO 170                                                         D 630
C                                                                       D 640
  150 DO 160 I=1,NA                                                     D 650
      IF (AWT(I,1).EQ.0.) GO TO 160                                     D 660
      DAWT(I)=IAWT                                                      D 670
      AWT(I,1)=ALOG(AWT(I,1))+.039                                      D 680
      AWT(I,3)=.20                                                      D 690
  160 CONTINUE                                                          D 700
C                                                                       D 710
  170 GO TO (240,180,180,200,220), IRT                                  D 720
C                                                                       D 730
  180 DO 190 I=1,NT                                                     D 740
      IF (RT(I,1).EQ.0.) GO TO 190                                      D 750
      DRT(I)=IRT                                                        D 760
      RT(I,2)=.500*RT(I,1)                                              D 770
      RT(I,3)=1.5*RT(I,1)                                               D 780
  190 CONTINUE                                                          D 790
      GO TO 240                                                         D 800
C                                                                       D 810
  200 DO 210 I=1,NT                                                     D 820
      IF (RT(I,1).EQ.0.) GO TO 210                                      D 830
      DRT(I)=IRT                                                        D 840
      RT(I,2)=0.                                                        D 850
      RT(I,3)=.200*RT(I,1)                                              D 860
  210 CONTINUE                                                          D 870
      GO TO 240                                                         D 880
C                                                                       D 890
  220 DO 230 I=1,NT                                                     D 900
      IF (RT(I,1).EQ.0.) GO TO 230                                      D 910
      DRT(I)=IRT                                                        D 920
      RT(I,1)=ALOG(RT(I,1))+.039                                        D 930
      RT(I,3)=.20                                                       D 940
  230 CONTINUE                                                          D 950
C                                                                       D 960
  240 RETURN                                                            D 970
      END                                                               D 980
      SUBROUTINE EZ
C
C
      INTEGER DW,DAWT,DRT
      INTEGER PATH,BEG,END
      DIMENSION ITRM(130)
      COMMON W(130,3),PN(130),AWT(515,3),PA(515),RT(20,3),N,NA,NT
      COMMON /C1/ ITN(20),NTYP(130),IMPP(130),II(515),JJ(515),MR(515)
      COMMON /C2/ IW,IAWT,IRT,DW(130),DAWT(515),DRT(20),CPI,ITERM,RSPNS,
     1TIME
      COMMON /C3/ NODE(130,4),IREG(65,2),N12,N12P,IR,LI,IM,IR2,LI2,IM2
      COMMON /C5/ IRUN,NPTH,PATH(690),BEG(30),END(30),PI(30),PI2(30)
C
      DO 10 M=1,NT
      J=ITN(M)
      ITRM(J)=M
   10 CONTINUE
C
      DO 120 K=1,IRUN
C
      DO 110 NP=1,NPTH
      IBEG=BEG(NP)
      IEND=END(NP)
      N2=PATH(IBEG)
      M=ITRM(N2)
      SR=SAMPL(DRT(M),RT(M,1),RT(M,2),RT(M,3))
      SW=SAMPL(DW(N2),W(N2,1),W(N2,2),W(N2,3))
      T=SW
      IF (T.LT.SR) GO TO 20
      J=1
      GO TO 40
   20 IB1=IBEG+1
C
      DO 30 J=IB1,IEND
      N1=N2
      N2=PATH(J)
      IR=NODE(N1,1)
      LI=NODE(N1,2)
      IM=IREG(IR,1)+(LI-1)*IREG(IR,2)-LI*(LI+1)/2
      IR2=NODE(N1,3)
      LI2=NODE(N1,4)
      IF (IR2.NE.0) IM2=IREG(IR2,1)+(LI2-1)*IREG(IR2,2)-LI2*(LI2+1)/2
      IAR=IJAD(N2)
      SAWT=SAMPL(DAWT(IAR),AWT(IAR,1),AWT(IAR,2),AWT(IAR,3))
      T=T+SAWT
      IF (T.GE.SR) GO TO 70
      SW=SAMPL(DW(N2),W(N2,1),W(N2,2),W(N2,3))
      T=T+SW
      IF (T.GE.SR) GO TO 40
   30 CONTINUE
```

```
C
      Z=0.0
      GO TO 100
C
C     LOCUS LIES IN NODE N2
   40 Z=PN(N2)
      IF (NTYP(N2).EQ.2) GO TO 50
      IF (SW.NE.0.0) Z=Z*(T-SR)/SW
      GO TO 60
   50 IF ((T-SR).LT.SW) Z=0.0
   60 IF (N2.EQ.PATH(IEND)) GO TO 100
      GO TO 80
C
C     LOCUS LIES IN ARC (N1,N2)
   70 Z=PA(IAR)*(T-SR)/SAWT
      Z=Z+(1.0-Z)*PN(N2)
      IF (N2.EQ.PATH(IEND)) GO TO 100
C
C     COMPUTE DETECTION PROBABILITY
   80 NXT=J+1
      DO 90 J=NXT,IEND
      N1=N2
      N2=PATH(J)
      IR=NODE(N1,1)
      LI=NODE(N1,2)
      IM=IREG(IR,1)+(LI-1)*IREG(IR,2)-LI*(LI+1)/2
      IR2=NODE(N1,3)
      LI2=NODE(N1,4)
      IF (IR2.NE.0) IM2=IREG(IR2,1)+(LI2-1)*IREG(IR2,2)-LI2*(LI2+1)/2
      IAR=IJAD(N2)
      Z=Z+(1.0-Z)*PA(IAR)
      Z=Z+(1.0-Z)*PN(N2)
   90 CONTINUE
C
  100 PI(NP)=PI(NP)+Z
      PI2(NP)=PI2(NP)+Z**2
  110 CONTINUE
C
  120 CONTINUE
C
      RETURN
      END
      FUNCTION IRG (I,J)
C
C
      INTEGER I,J
      COMMON /C3/ NODE(130,4)
      IF (NODE(I,1).NE.NODE(J,1)) GO TO 10
      IRG=NODE(I,1)
      GO TO 50
   10 IF (NODE(I,1).NE.NODE(J,3)) GO TO 20
      IRG=NODE(I,1)
      GO TO 50
   20 IF (NODE(I,3).NE.NODE(J,1)) GO TO 30
      IRG=NODE(I,3)
      GO TO 50
   30 IF (NODE(I,3).NE.NODE(J,3)) GO TO 40
      IRG=NODE(I,3)
      GO TO 50
   40 CONTINUE
      IRG=0
   50 CONTINUE
      RETURN
      END
      SUBROUTINE MNDPT1
C
C
      DIMENSION TLBL(130), IMPT(130), SW(130), SAWT(515)
      LOGICAL FLAG,MFLAG
      INTEGER DW,DAWT,DRT
      COMMON W(130,3),PN(130),AWT(515,3),PA(515),RT(20,3),N,NA,NT
      COMMON /C1/ ITN(20),NTYP(130),IMPP(130),II(515),JJ(515),MR(515)
      COMMON /C2/ IW,IAWT,IRT,DW(130),DAWT(515),DRT(20),CPI,ITERM,RSPNS,
     1TIME
      COMMON /C3/ NODE(130,4),IREG(65,2),N12,N12P,IR,LI,IM,IR2,LI2,IM2
      COMMON /C4/ PLBL(130),ITEMP(130),IPERM(130),BIG
      COMMON /C6/ IMP(130)
      COMMON /C7/ MFLAG
C
C     RESTORE PROBABILITY PREDECESSORS IN IMP
      DO 10 I=1,N
   10 IMP(I)=IMPP(I)
C
C     SET SAMPLE VALUES TO -1 SO THAT THE PROGRAM CAN LATER DETERMINE
C     WHETHER VALUES HAVE BEEN SAMPLED YET DURING A GIVEN RUN.
      DO 20 I=1,N
   20 SW(I)=-1.0
      DO 30 J=1,NA
   30 SAWT(J)=-1.0
C
C     FOR EACH TERMINAL NODE DO THE FOLLOWING:
C     1. EXECUTE DIJKSTRA-YEN SEARCH FOR LEAST TIME PATHS FROM TERMINAL
C      NODE. TERMINATE WHEN ALL TEMPORARY LABELS EXCEED THE RESPONSE
C      TIME, OR WHEN A LOCUS NODE BECOMES PERMANENTLY LABELLED.
C     2. IN THE LATTER CASE A LOCUS POINT LIES OFF-SITE SO THE
C      LEAST-DETECTION-PROBABILITY IS 0. STORE THE MINIMUM TIME PATH
C      FROM THE BOUNDARY AND RETURN TO PATHS.
C     3. IN THE FORMER CASE, DETERMINE LOCUS POINTS AND THE CORRESPONDIN
C      LEAST-DETECTION-PROBABILITIES, ALWAYS UPDATING THE MINIMUM.
C      IF THE MINIMUM EVER HITS 0 STORE THE MINIMUM TIME PATH AND
C      RETURN TO PATHS.
C
      CPI=1.
      TIME=BIG
      DO 220 M=1,NT
C
C     INITIALIZE.
      FLAG=.FALSE.
      ITRM=ITN(M)
      IMPT(ITRM)=0
      R=SAMPL(DRT(M),RT(M,1),RT(M,2),RT(M,3))
      DO 40 I=1,N
      TLBL(I)=BIG
      ITEMP(I)=I
   40 CONTINUE
      L=0
      K=N
      IF (SW(ITRM).LT.0.) SW(ITRM)=SAMPL(DW(ITRM),W(ITRM,1),W(ITRM,2),W(
     1ITRM,3))
      WT=SW(ITRM)
      TLBL(ITRM)=WT
      IF (WT.GE.R) GO TO 90
      IP=ITRM
      IQ=ITRM
C
C     PERMANENTLY LABEL NODE IP.
   50 L=L+1
      IPERM(L)=IP
      ITEMP(IQ)=ITEMP(K)
      K=K-1
      IF (IP.LE.N12) GO TO 60
C
C     A BOUNDARY NODE IS INSIDE THE LOCUS
      IF (TLBL(IP).GE.TIME) GO TO 210
      TIME=TLBL(IP)
      RSPNS=R
      LN=IP
      LNP=IMPT(LN)
      CPI=0.
      FLAG=.TRUE.
      GO TO 190
C
   60 I=IP
      IR=NODE(I,1)
      LI=NODE(I,2)
      IM=IREG(IR,1)+(LI-1)*IREG(IR,2)-LI*(LI+1)/2
      IR2=NODE(I,3)
      LI2=NODE(I,4)
      IF (IR2.NE.0) IM2=IREG(IR2,1)+(LI2-1)*IREG(IR2,2)-LI2*(LI2+1)/2
      V=BIG
C
C     TREAT EACH TEMPORARILY LABELED NODE.
C     V IS THE SMALLEST SUCH LABEL.
      DO 80 IT=1,K
      J=ITEMP(IT)
      IAR=IJAD(J)
      IF (IAR.EQ.0) GO TO 70
      IF (SAWT(IAR).LT.0.) SAWT(IAR)=SAMPL(DAWT(IAR),AWT(IAR,1),AWT(IAR,
     12),AWT(IAR,3))
      DIJ=SAWT(IAR)
      IF (SW(J).LT.0.) SW(J)=SAMPL(DW(J),W(J,1),W(J,2),W(J,3))
      Z=TLBL(I)+DIJ+SW(J)
      IF (Z.GE.TLBL(J)) GO TO 70
      TLBL(J)=Z
      IMPT(J)=I                                                         G1010
   70 IF (TLBL(J).GE.V) GO TO 80                                        G1020
      V=TLBL(J)                                                         G1030
      IP=J                                                              G1040
      IQ=IT                                                             G1050
   80 CONTINUE                                                          G1060
      IF (V.LT.R) GO TO 50                                              G1070
      IF (CPI.EQ.0.) GO TO 210                                          G1080
      GO TO 120                                                         G1090
C                                                                       G1100
C     LOCUS LIES INSIDE THE TERMINAL NODE ITN(M).                       G1110
   90 Z=PN(ITRM)                                                        G1120
      IF (NTYP(ITRM).EQ.2) GO TO 100                                    G1130
      IF (WT.NE.0.) Z=Z*(WT-R)/WT                                       G1140
      GO TO 110                                                         G1150
  100 IF (R.GT.0.) Z=0.                                                 G1160
  110 Z=Z+(1.0-Z)*PLBL(ITRM)                                            G1170
      IF (Z.GE.CPI) GO TO 210                                           G1180
      CPI=Z                                                             G1190
      ITERM=ITRM                                                        G1200
      GO TO 210                                                         G1210
C                                                                       G1220
C     ALL NODES INSIDE THE LOCUS ARE PERMANENTLY LABELED.               G1230
C     IDENTIFY THE LOCUS NODES AND ASSOCIATED LOCUS POINTS.             G1240
C                                                                       G1250
  120 DO 180 IT=1,K                                                     G1260
      J=ITEMP(IT)                                                       G1270
      IF (TLBL(J).EQ.BIG) GO TO 180                                     G1280
C                                                                       G1290
C     J IS A LOCUS NODE                                                 G1300
      IF (PLBL(J).GE.CPI) GO TO 180                                     G1310
      DIF=TLBL(J)-R                                                     G1320
      WJ=SW(J)                                                          G1330
      IF (DIF.GT.WJ) GO TO 150                                          G1340
C                                                                       G1350
C     THE ASSOCIATED LOCUS POINT LIES INSIDE J.                         G1360
      JP=IMPT(J)                                                        G1370
      Z=PN(J)                                                           G1380
      IF (NTYP(J).EQ.2) GO TO 130                                       G1390
      IF (WJ.NE.0.) Z=Z*DIF/WJ                                          G1400
      GO TO 140                                                         G1410
  130 IF (DIF.LT.WJ) Z=0.                                               G1420
  140 Z=Z+(1.-Z)*PLBL(J)                                                G1430
      GO TO 170                                                         G1440
C                                                                       G1450
C     THE ASSOCIATED LOCUS POINT LIES ON AN ARC FROM J.                 G1460
  150 X=PLBL(J)                                                         G1470
      X=X+(1.-X)*PN(J)                                                  G1480
      IF (X.GE.CPI) GO TO 180                                           G1490
      IR=NODE(J,1)                                                      G1500
      LI=NODE(J,2)
      IM=IREG(IR,1)+(LI-1)*IREG(IR,2)-LI*(LI+1)/2
      IR2=NODE(J,3)
      LI2=NODE(J,4)
      IF (IR2.NE.0) IM2=IREG(IR2,1)+(LI2-1)*IREG(IR2,2)-LI2*(LI2+1)/2
      Z=BIG
      DO 160 IS=1,L
      I=IPERM(IS)
      IAR=IJAD(I)
      IF (IAR.EQ.0) GO TO 160
      DIJ=SAWT(IAR)
      V=PA(IAR)*(TLBL(I)+DIJ-R)/DIJ
      IF (V.GE.Z) GO TO 160
      Z=V
      JP=I
  160 CONTINUE
      Z=Z+(1.-Z)*X
C
  170 IF (Z.GE.CPI) GO TO 180
      FLAG=.TRUE.
      CPI=Z
      LN=J
      LNP=JP
  180 CONTINUE
C
  190 IF (.NOT.FLAG) GO TO 210
      ITERM=ITRM
      IMP(LNP)=LN
      I=LNP
  200 IP=IMPT(I)
      IF (IP.EQ.0) GO TO 210
      IMP(IP)=I
      I=IP
      GO TO 200
C
  210 CONTINUE
      IF (CPI.EQ.0.0.AND.MFLAG) GO TO 230
  220 CONTINUE
C
  230 RETURN
      END
      SUBROUTINE MOD (I,X1,X2,X3,IDSTB)
C
C
      GO TO (10,20,30,40,50), IDSTB
C
   10 CONTINUE
      WRITE (6,120) X1
      GO TO 60
   20 CONTINUE
      WRITE (6,130) X2,X3
      GO TO 60
   30 CONTINUE
      WRITE (6,140) X1,X2,X3
      GO TO 60
   40 CONTINUE
      WRITE (6,150) X1,X3
      GO TO 60
   50 XM=EXP(X1-X3**2)
      STDV=XM*EXP(1.5*X3**2)*SQRT(EXP(X3**2)-1.0)
      WRITE (6,160) XM,STDV
      GO TO 60
C
C
   60 CONTINUE
      WRITE (6,170)
      READ * , ID
      IF (EOF(15).NE.0.0) GO TO 60
      IF (ID.EQ.0) RETURN
      IDSTB=ID
      IF (I.EQ.2.OR.I.EQ.4) IDSTB=1
      GO TO (70,80,90,100,110), IDSTB
C
   70 CONTINUE
      WRITE (6,180)
      READ * , X1
      IF (EOF(15).NE.0.0) GO TO 70
      RETURN
   80 CONTINUE
      WRITE (6,190)
      READ * , X2,X3
      IF (EOF(15).NE.0.0) GO TO 80
      IF (X2.LT.X3) RETURN
      WRITE (6,200)
      GO TO 80
   90 CONTINUE
      WRITE (6,210)
      READ * , X1,X2,X3
      IF (EOF(15).NE.0.0) GO TO 90
      IF (X1.GT.X2.AND.X3.GT.X1) RETURN
      WRITE (6,220)
      WRITE (6,230)
      GO TO 90
  100 CONTINUE
      WRITE (6,240)
      READ * , X1,X3
      IF (EOF(15).NE.0.0) GO TO 100
      X2=0.0
      RETURN
  110 CONTINUE
      WRITE (6,250)
      READ * , X1
      IF (EOF(15).NE.0.0) GO TO 110
      X3=.20
      X1=ALOG(X1)+(X3**2)
      RETURN
C
C
C
  120 FORMAT (8X,21HDETERMINISTIC VALUE =,F9.4/)
  130 FORMAT (8X,15HUNIFORM   MIN =,F9.4,8H   MAX =,F9.4/)
  140 FORMAT (8X,19HTRIANGULAR   MODE =,F9.4,8H   MIN =,F9.4,8H   MAX =,
     1F9.4/)
  150 FORMAT (8X,15HNORMAL   MEAN =,F9.4,9H   STDV =,F9.4/)
  160 FORMAT (8X,18HLOGNORMAL   MODE =,F9.4,9H   STDV =,F9.4/)
  170 FORMAT (25HENTER: DISTRIBUTION CODE )
  180 FORMAT (27HENTER: DETERMINISTIC VALUE )
  190 FORMAT (34HENTER UNIFORM PARAMETERS: MIN MAX )
  200 FORMAT (25HMIN MUST BE LESS THAN MAX)
  210 FORMAT (42HENTER TRIANGULAR PARAMETERS: MODE MIN MAX )
  220 FORMAT (26HMIN MUST BE LESS THAN MODE)
  230 FORMAT (30HAND MODE MUST BE LESS THAN MAX)
  240 FORMAT (35HENTER NORMAL PARAMETERS: MEAN STDV )
  250 FORMAT (32HENTER LOGNORMAL PARAMETER: MODE )
      END


      SUBROUTINE NDNUM (N,NDNAM,J)
C
C
      DIMENSION NDNAM(1)
      DO 10 I=1,N
      IF (NDNAM(I).EQ.J) GO TO 20
   10 CONTINUE
      WRITE (6,30) J
      J=0
      RETURN
   20 J=I
      RETURN
C
C
C
   30 FORMAT (I11,20H IS NOT A NODE LABEL)
      END
      SUBROUTINE PATHS
C
C
      INTEGER PATH,BEG,END
      INTEGER DW,DAWT,DRT,HEAD,Z
      DIMENSION NDNAM(130)
      REAL MN,MMM
      LOGICAL FLAG,MFLAG
      COMMON W(130,3),PN(130),AWT(515,3),PA(515),RT(20,3),N,NA,NT
      COMMON /C1/ ITN(20),NTYP(130),IMPP(130),II(515),JJ(515),MR(515)
      COMMON /C2/ IW,IAWT,IRT,DW(130),DAWT(515),DRT(20),CPI,ITERM,RSPNS
     1TIME
      COMMON /C3/ NODE(130,4),IREG(65,2),N12,N12P,IR,LI,IM,IR2,LI2,IM2
      COMMON /C4/ PLBL(130),ITEMP(130),IPERM(130),BIG
      COMMON /C5/ IRUN,NPTH,PATH(690),BEG(30),END(30),UM(30),UM2(30)
      COMMON /C6/ IMP(130)
      COMMON /C7/ MFLAG
      COMMON /C8/ HEAD(5),Z(130),PCNT(30),DUMP,MRUN,ISEED,PAI,DM,DM2
      COMMON /C9/ NCNT(130),CM(30),CM2(30),UMIN
      COMMON /C10/ PAI2
      DATA MAXNP/30/
C
C
      CALL CLRY (CM,30)
      CALL CLRY (CM2,30)
      CALL CLRY (NCNT,130)
      CALL CLRY (PCNT,30)
C
C
C     RUN THE PHASE 1 PATH GENERATING ALGORITHM
C     IF MINIMIZATION MEASURE IS INTERRUPTION PROBABILITY
C     RUN THE DIJKSTRA PROBABILITY SEARCH
      IF (MFLAG) CALL PDIJK
C
C     RUN THE STOCHASTIC MINDPT SEARCH
C     FOR EACH RUN COLLECT THE APPROPRIATE TIME OR INTERRUPTION
C     STATISTICS. SAVE NEW PATHS AS THEY ARE GENERATED UNLESS
C     THE MAXIMUM ALLOWABLE NUMBER OF PATHS OR TOTAL LINEAR
C     STORAGE OF PATH IS EXCEEDED. IN THIS CASE CONTINUE TO
C     COLLECT STATISTICS IN THE DUMP.
C
      PAI2=0
      DM=0
      DM2=0
      PAI=0
      DUMP=0
      NPTH=0
      FLAG=.TRUE.
      DO 90 K=1,MRUN
      CALL MNDPT1
      IF (MFLAG) MMM=CPI
      IF (.NOT.MFLAG) MMM=TIME
      PAI=PAI+MMM
      PAI2=PAI2+(MMM**2)
      IF (K.EQ.1) GO TO 50
      CALL CHKPTH (ITERM)
C
C     TRACE CURRENT PATH FROM TERMINAL NODE TO BOUNDARY
C     USING PREDECESSOR VECTOR.
      DO 20 NP=1,NPTH
      IBEG=BEG(NP)
      IEND=END(NP)
      NOD=ITERM
      DO 10 J=IBEG,IEND
      IF (PATH(J).NE.NOD) GO TO 20
   10 NOD=IMP(NOD)
C
C     CURRENT PATH IS PATH NP
      PCNT(NP)=PCNT(NP)+1.0
      CM(NP)=CM(NP)+MMM
      CM2(NP)=CM2(NP)+(MMM**2)
      GO TO 90
   20 CONTINUE
C
C     CURRENT PATH IS A NEW ONE
      IF (.NOT.FLAG) GO TO 30
      J=END(NPTH)
      GO TO 60
C
C     DUMP CURRENT PATH
   30 DUMP=DUMP+1.0
      NOD=ITERM
   40 NCNT(NOD)=NCNT(NOD)+1
      NOD=IMP(NOD)
      IF (NOD.NE.0) GO TO 40
      DM=DM+MMM
      DM2=DM2+(MMM**2)
      GO TO 90
C
C     STORE CURRENT PATH
   50 J=0
   60 BEG(NPTH+1)=J+1
      NOD=ITERM
   70 IF (J.EQ.500) GO TO 80
      J=J+1
      PATH(J)=NOD
      NOD=IMP(NOD)
      IF (NOD.NE.0) GO TO 70
C
C     CURRENT PATH HAS BEEN STORED
```

```
      NPTH=NPTH+1
      IF (NPTH.EQ.MAXNP) GO TO 80
      CM(NPTH)=MMM
      CM2(NPTH)=MMM**2
      PCNT(NPTH)=1
      END(NPTH)=J
      GO TO 90
C
C     STORAGE LIMIT OF PATH EXCEEDED
   80 FLAG=.FALSE.
      GO TO 30
   90 CONTINUE
C
C
      RETURN
      END



      SUBROUTINE PAWZ
C
C
      WRITE (6,10)
      WRITE (6,20)
      READ 30, I
      Z=EOF(15)
      WRITE (6,40)
      RETURN
C
C
C
   10 FORMAT (1H )
   20 FORMAT (/43H CLEAR SCREEN AND PRESS RETURN TO CONTINUE )
   30 FORMAT (A1)
   40 FORMAT (1H )
      END
      SUBROUTINE PDIJK
C
C
      COMMON W(130,3),PN(130),AWT(515,3),PA(515),RT(20,3),N,NA,NT
      COMMON /C1/ ITN(20),NTYP(130),IMPP(130),II(515),JJ(515),MR(515)
      COMMON /C2/ IW,IAWT,IRT,DW(130),DAWT(515),DRT(20),CPI,ITERM,RSPNS,
     1TIME
      COMMON /C3/ NODE(130,4),IREG(65,2),N12,N12P,IR,LI,IM,IR2,LI2,IM2
      COMMON /C4/ PLBL(130),ITEMP(130),IPERM(130),BIG
C
C     DIJKSTRA-YEN SEARCH FOR LEAST-DETECTION-PROBABILITY PATHS FROM
C     OFF-SITE.
C     PLBL(I) IS THE  MINIMUM PROBABILITY OF DETECTION UP TO,
C     BUT NOT INCLUDING NODE I.
C     IMPP(I) IS THE IMMEDIATE PREDECESSOR OF NODE I ALONG THE
C     LEAST-DETECTION-PROBABILITY PATH FROM OFF-SITE.
C
C     INITIALIZE - BOUNDARY NODES GET 0 LABELS, ALL OTHERS INFINITE
      DO 10 I=1,N12
      PLBL(I)=BIG
      ITEMP(I)=I
   10 CONTINUE
      DO 20 I=N12P,N
      PLBL(I)=0.
      ITEMP(I)=I
   20 CONTINUE
      L=0
      K=N
      IP=N
      IQ=N
C
C     PERMANENTLY LABEL NODE IP.
C     L IS THE NUMBER OF PERMANENTLY LABELLED NODES
C     K IS THE NUMBER OF TEMPORARILY LABELLED NODES
   30 L=L+1
      IPERM(L)=IP
      ITEMP(IQ)=ITEMP(K)
      K=K-1
      IF (K.EQ.0) RETURN
      I=IP
      IR=NODE(I,1)
      LI=NODE(I,2)
      IM=IREG(IR,1)+(LI-1)*IREG(IR,2)-LI*(LI+1)/2
      IR2=NODE(I,3)
      LI2=NODE(I,4)
      IF (IR2.NE.0) IM2=IREG(IR2,1)+(LI2-1)*IREG(IR2,2)-LI2*(LI2+1)/2
      Y=PLBL(I)+(1.0-PLBL(I))*PN(I)
      YC=1.0-Y
      V=BIG
C
```

```
C     TREAT EACH TEMPORARILY LABELED NODE.
C     V IS THE SMALLEST SUCH LABEL.
      DO 50 IT=1,K
      J=ITEMP(IT)
      IAR=IJAD(J)
      IF (IAR.EQ.0) GO TO 40
      DIJ=PA(IAR)
      Z=Y+YC*DIJ
      IF (Z.GE.PLBL(J)) GO TO 40
      PLBL(J)=Z
      IMPP(J)=I
   40 IF (PLBL(J).GE.V) GO TO 50
      V=PLBL(J)
      IP=J
      IQ=IT
   50 CONTINUE
      GO TO 30
      END
      SUBROUTINE PRNT (Z,NCNT,FMRN,IBEG,IEND,JEND,NDNAM)
C
C
      DIMENSION NCNT(1)
      REAL NCNT
      INTEGER Z(1),NDNAM(1)
      DIMENSION N(2), R(2)
      WRITE (6,60)
C
      M=JEND-IBEG+1
      I=0
C
C
      DO 30 J=1,M
      L=IBEG+J-1
      IZL=Z(L)
      NI=NDNAM(IZL)
      IF (NI.GE.1000) GO TO 20
      I=I+1
      N(I)=NI
      R(I)=NCNT(L)/FMRN
      IF (I.LT.2) GO TO 10
      WRITE (6,70) (N(K),R(K),K=1,2)
      WRITE (6,60)
C
      I=0
C
   10 CONTINUE
   20 CONTINUE
   30 CONTINUE
      IF (I.LE.0) GO TO 50
      IF (I.EQ.1) GO TO 40
      WRITE (6,70) (N(J),R(J),J=1,2)
      WRITE (6,60)
      GO TO 50
   40 CONTINUE
      WRITE (6,70) N(1),R(1)
      WRITE (6,60)
      GO TO 50
   50 CONTINUE
      RETURN
C
C
C
   60 FORMAT (2H *,51X,1H*)
   70 FORMAT (I9,6X,F4.2,I16,7X,F4.2)
      END
      SUBROUTINE SORT (X,Z,IBEG,IEND,JEND,TOL,EPS)
C
C
      INTEGER Z(1)
      INTEGER ZMAX
      DIMENSION X(1)
C
C     Z(K) TELLS WHICH ORIGINAL POSITION CORRESPONDS TO K AFTER SORT
C
      CUM=0.0
      DO 10 K=IBEG,IEND
   10 Z(K)=K
C
      L=IEND-IBEG
      DO 30 I=1,L
      J=IBEG+I-1
      J1=J+1
      XMAX=X(J)
      ZMAX=J
C
      DO 20 K=J1,IEND
      IF (X(K).LE.XMAX) GO TO 20
      XMAX=X(K)
      ZMAX=K
   20 CONTINUE
C
      IF (XMAX.LE.EPS) GO TO 50
C
      X(ZMAX)=X(J)
      X(J)=XMAX
      ITEMP=Z(ZMAX)
      Z(ZMAX)=Z(J)
      Z(J)=ITEMP
C
      CUM=CUM+XMAX
      IF (CUM.GE.TOL) GO TO 40
   30 CONTINUE
C
      JEND=IEND
      RETURN
C
   40 JEND=J
      RETURN
C
   50 JEND=J-1
      RETURN
      END
      SUBROUTINE THYME (T,TDEV,W,AWT,DW,DAWT,BEG,END,PATH,NPTH)
C
C
      REAL MU
      INTEGER PATH(1),BEG(1),END(1),DW(1),DAWT(1)
      DIMENSION W(130,3), AWT(515,3), T(515), TDEV(1)
      COMMON /C3/ NODE(130,4),IREG(65,2),N12,N12P,IR,LI,IM,IR2,LI2,IM2
C
C
      DO 170 NP=1,NPTH
      IBEG=BEG(NP)
      IEND=END(NP)
      N2=PATH(IBEG)
      IB1=IBEG+1
      GO TO (10,20,30,10,40), DW(N2)
   10 MU=W(N2,1)
      GO TO 50
   20 MU=(W(N2,2)+W(N2,3))/2.0
      GO TO 50
   30 MU=(W(N2,1)+W(N2,2)+W(N2,3))/3.0
      GO TO 50
   40 MU=EXP(W(N2,1)+.0195)
   50 V=VAR(DW(N2),MU,W(N2,2),W(N2,3))
C
      DO 160 J=IB1,IEND
      N1=N2
      N2=PATH(J)
      IR=NODE(N1,1)
      LI=NODE(N1,2)
      IM=IREG(IR,1)+(LI-1)*IREG(IR,2)-LI*(LI+1)/2
      IR2=NODE(N1,3)
      LI2=NODE(N1,4)
      IF (IR2.NE.0) IM2=IREG(IR2,1)+(LI2-1)*IREG(IR2,2)-LI2*(LI2+1)/2
      IAR=IJAD(N2)
C
      GO TO (60,70,80,60,90), DAWT(IAR)
   60 X=AWT(IAR,1)
      GO TO 100
   70 X=(AWT(IAR,2)+AWT(IAR,3))/2.0
      GO TO 100
   80 X=(AWT(IAR,1)+AWT(IAR,2)+AWT(IAR,3))/3.0
      GO TO 100
   90 X=EXP(AWT(IAR,1)+.0195)
  100 MU=MU+X
      V=V+VAR(DAWT(IAR),X,AWT(IAR,2),AWT(IAR,3))
      GO TO (110,120,130,110,140), DW(N2)
  110 X=W(N2,1)
      GO TO 150
  120 X=(W(N2,2)+W(N2,3))/2.0
      GO TO 150
  130 X=(W(N2,1)+W(N2,2)+W(N2,3))/3.0
      GO TO 150
  140 X=EXP(W(N2,1)+.0195)
  150 MU=MU+X
      V=V+VAR(DW(N2),X,W(N2,2),W(N2,3))
  160 CONTINUE
C
      T(NP)=MU
      TDEV(NP)=SQRT(V)
  170 CONTINUE
      RETURN
      END


      FUNCTION IJAD (J)
C
C
      COMMON /C3/ NODE(130,4),IREG(65,2),N12,N12P,IR,LI,IM,IR2,LI2,IM2
C     I IS AN IMPLICIT ARGUMENT SINCE IR,LI,IM,IR2,IM2 ARE
C     DETERMINED BY I AND ARE SET BEFORE CALLING IJAD.  IF NODES I
C     J ARE NOT ADJACENT, THEN IJAD=0.  OTHERWISE, IJAD IS THE ADDRESS
C     IN THE ARC LIST.
      IF (NODE(J,1).NE.IR) GO TO 10
      LJ=NODE(J,2)
      GO TO 20
   10 IF (NODE(J,3).NE.IR) GO TO 40
      LJ=NODE(J,4)
   20 IF (LI.GT.LJ) GO TO 30
      IJAD=IM+LJ
      RETURN
   30 IJAD=IREG(IR,1)+(LJ-1)*IREG(IR,2)-LJ*(LJ+1)/2+LI
      RETURN
   40 IF (NODE(J,1).NE.IR2) GO TO 50
      LJ=NODE(J,2)
      GO TO 60
   50 IF (NODE(J,3).NE.IR2) GO TO 80
      IF (IR2.EQ.0) GO TO 80
      LJ=NODE(J,4)
   60 IF (LI2.GT.LJ) GO TO 70
      IJAD=IM2+LJ
      RETURN
   70 IJAD=IREG(IR2,1)+(LJ-1)*IREG(IR2,2)-LJ*(LJ+1)/2+LI2
      RETURN
   80 IJAD=0
      RETURN
      END
      FUNCTION RNORM (X1,X3)
C
C
      SUM=0.0
      DO 10 I=1,12
   10 SUM=SUM+RANF(N)
      V=SUM-6.0
      RNORM=V*X3+X1
      RETURN
      END
```

```
      FUNCTION SAMPL (IDSTB,X1,X2,X3)
C
C
      GO TO (10,20,30,50,60), IDSTB
C
C     DETERMINISTIC
   10 SAMPL=X1
      RETURN
C
C     UNIFORM   X2-MIN  X3-MAX
   20 SAMPL=X2+(X3-X2)*RANF(N)
      RETURN
C
C     TRIANGULAR   X1-MODE  X2-MIN  X3-MAX
   30 RN=RANF(N)
      Z1=X1-X2
      Z3=X3-X2
      IF (RN.GE.Z1/Z3) GO TO 40
      SAMPL=X2+SQRT(Z1*Z3*RN)
      RETURN
   40 CONTINUE
      OP=(X3-X1)*Z3*(1.0-RN)
      SAMPL=X3-SQRT(OP)
      RETURN
C
C     NORMAL   X1-MEAN  X3-STDV
   50 SAMPL=RNORM(X1,X3)
      IF (SAMPL.GE.0.0) RETURN
      SAMPL=0.0
      RETURN
C
C     LOGNORMAL   X1 AND X3 ARE THE CORRESPONDING NORMAL PARAMETERS
   60 SAMPL=EXP(RNORM(X1,X3))
      RETURN
      END
      FUNCTION VAR (IDSTB,X1,X2,X3)
C
C
      GO TO (10,20,30,40,50), IDSTB
C
   10 VAR=0.0
      RETURN
   20 VAR=((X3-X2)**2)/12.
      RETURN
   30 VAR=(X2*(X2-X1)+X3*(X3-X2)+X1*(X1-X3))/18.
      RETURN
   40 VAR=X3**2
      RETURN
   50 Z=X3**2
      VAR=X1**2*(EXP(Z)-1.0)
      RETURN
      END
```

1P2

BLANK PAGE

INDEX

Adversary strategy, 20
Analytic evaluation, 37, 46, 47
Analytic technique, 15, 23
Arc, 17
  delay time, 36
  detection probability for, 17, 19, 21, 65
  labels, 17, 19, 53
  traversal time, 19, 21, 33
  traversal time, modification of, 64
  weights, 17, 41, 53
Arithmetic mean, 26, 44, 46, 47, 50

Binomial trials, 23

Central processor (CP) time, 67, 93, 98
Coefficient of variation, 50
  values for, 37, 38
Computation, cost of, 93
Conditional probability, 25-26
  estimate of mean, 49, 73, 74
  evaluation of, 26, 43, 44
  expected value for, 26
  NOT MIN, 27, 49, 74, 79, 83
Criticality estimates, 25, 26, 27, 47, 49, 51, 74, 79, 83, 87
  independent of seed and MRUN, 90
Critical path, 14, 15, 20, 25, 26, 79, 83, 87
  contention for, 79, 87
  overlooking, probability of, 22, 32, 83

Default data, 54
Default values, 19, 22
Detection probability, 14, 25
  cumulative, 39-41, 43
  examination of, 64, 65
  fractional contribution to, 44
  initial values as means, 19, 22
  as measure, 20, 21
  modification of, 64, 65
  and precision, 75
  set to zero, 36
  weighting, 40
  weights computed, 19

DFLT, 36-37, 64
Dijkstra search, 20, 42, 93
Distribution, characterizing quantities, 37, 65
  codes, 36, 54, 63-66, 75
  deterministic, 19, 25, 36, 37, 41, 63, 64
  lognormal, 37, 38
  modification of, 64
  normal, 37
  null, 38, 66
  symmetrical, 37, 65
  triangular, 37, 63, 64, 75
  uniform, 37
Distribution-dependent quantities, 37
Dump, 39, 49
  count, 39
  nodes in, 75
  rank, 49, 74

EASI, 15
Error, fatal, 66
  input, 62
  Type I, 90
  Type II, 30
Estimates, for dump, 39, 49, 74
  examples of, 28-31
  facility, 23, 73, 74
  path criticality, 25, 26-27, 49, 74, 87
  Phase 1 and Phase 2, 23-25, 26, 49
  for unconditional mean, 79, 87
EZ, 33, 37, 46-48

Facility estimates, 23, 47, 48, 49, 73
  independent of seed and MRUN, 90
Facility graph, 17-19, 33
Facility minimum (min P), 23, 28
  variance of, 30
Facility performance, 14, 23, 25
  dependence on measure, 79
  indeterminacy, 24, 30
  and security, 24, 30, 31
Facility size, 62, 98
  and run time, 93
  and storage space, 99
Flow diagram, 33

Fractional dump count, 39
Generated path, 14, 15, 20, 25, 39
  and sample size, 87
  and seed, 83, 87
Guard response time, 20, 36, 41
  set to large value (BIG), 21, 36, 41, 66
  set to zero, 21

Halt procedure, 67
HEADER (title), 63, 65

Information, dumped, 38
  node ranking, 75
  path ranking, 73-74
  saved, 38, 48
  storing, 38-39
  summary, 67
Input, 33
  brief mode, 33, 54
  echoed, 36, 67
  error, 62
  extended mode, 33, 54
  file, 19, 33, 53
  prompt for, 54, 67
  user-specified, 33, 36-38, 53, 54-67
  values, modification of, 38
Interruption probability, 14, 25
  conditional, 25, 49
  conditional, computation of, 44-46
  conditional, estimate of, 26, 49, 73
  as measure, 20, 25, 41, 46, 47
  unconditional, 26, 27, 47, 49
  unconditional, estimate of, 26, 48, 49, 74
Interval, 30
  and MRUN, 92
  and precision, 30, 75
  and sample size, 83
  and security system, 31
IRUN, 23, 25, 48, 93, 94
  fixed internally, 66
  recommended value, 32, 92
  and sample size, 32

JOB ACTIVE, 67

Linear correlation coefficient, 27, 28, 49, 51
  indeterminate, 74
  not unity, 74
  values for, 27
Locus point, 20, 43
Lower bound, 14, 46, 48, 73, 90
  check on sample size, 24, 25
  expected value for, 26
  greater than upper bound, 24, 31, 83
  LB, 23, 24, 26, 28

Measure(s), 13, 20-21
  and adversary strategy, 20
  available, 14
  as input, 62
  minimum detection probability, 14
  minimum interruption probability, 14, 33
  minimum traversal time, 14, 33
  and path evaluation, 46
  and run time, 94, 99
MINDPT, 15
MNDPT1, 33, 37, 41-44
  and run time, 98
MOD, 37, 38
Mode (most probable value), 19, 37, 65
  equal to BIG, 38
Monte Carlo approach, 14, 47
Monte Carlo sampling, 15, 22, 42
Monte Carlo technique, 13, 15, 22
  and path evaluation, 22, 46
MRUN, 23, 25, 28, 48, 90, 92
  and LB > MIN, 92
  less than IRUN, 25
  and paths generated, 87, 92
  and precision estimates, 83, 92
  realizations, 22, 41
  recommended value for, 32, 92
  and run time, 32, 92
  and sample size, 31-32

Node, 17
  barrier, 17, 62n, 65
  boundary, 17, 40, 62
  count, 39, 48
  detection probability for, 17, 21
  detection type (NTYP), 33, 36, 44, 62
  labels, 44, 53, 62n
  locus, 43
  predecessor, 39, 40, 41, 43
  rank, 14, 25, 49, 51, 75
  target, 41, 43, 62
  terminal, 39, 40
  type, 62
  traversal time, 17, 21, 33, 36
  weights, 19, 53
NOS, 53, 67
NOT MIN (derived), 27, 49n

Output, 33, 48, 67
  extended content, 49, 67, 73, 79
  node ranking, 75

  path description, 73
  path ranking, 73-74
  PATHS summary, 67, 73
  standard content, 49, 67, 73, 79, 92

Path, composite, 23
  count, 39, 50, 51
  criticality, 14, 25, 26, 79, 83, 87, 92
  description, 14, 44, 48, 73
  dumped, 39, 49, 67
  evaluation phase (Phase 2), 22-23, 46-48
  generated, 38, 50, 53, 69
  generation phase (Phase 1), 22-23, 39-46
  minimum detection, 21, 39
  minimum traversal time, 21, 41-42
  nodal sequence for, 64, 73
  overlooking, probability of 22, 32
  rank, 14, 26, 28, 49, 51, 74, 90, 92
  real, 23, 74
  saved, 48, 49, 51, 67, 93
  segment, 13, 20, 42
Pathfinding algorithm, 20
Pathfinding search, least detection probability, 39-41
  least traversal time, 41-44
PATHS (subroutine), 33, 38-39, 44-46
PDIJK, 33, 39-41
  and run time, 98
Precision, 28, 30, 31, 32
  estimates of, 47, 48, 73, 83
  and interval, 30, 75
  and sample size, 30, 32, 83
Prior probability distribution(s), 15, 22, 47, 75
  assignment of, 37
  characterizing quantities, 37, 65
  list of, 37
  modification of, 38
  Monte Carlo simulation in, 37
  and path criticality, 87n
Probabilistic models, 101
Procedure file, access to, 53-54
Pseudo-node, 64

Random number, 37, 63
  seed, 36, 63, 63n
Random variables, 13, 22, 23
Realization, 22, 41
Recurrence relation, 40, 42
Replication, 22, 25, 48
Response time, 21, 36, 43, 44, 75
  and BIG, 36, 38, 41, 43, 66
  distribution, 38, 41, 66
  as input, 63
  locus, 20, 36, 41, 42
  not specified, 66
  and precision, 75
  and run time, 93, 94, 98
  values for, 38, 41
Run, 22
Run time, 93-99
  dependence on measure, 94, 99
  estimates of, 99
  and facility size, 93, 99
  Phase 1 contribution to, 93
  Phase 2 contribution to, 93, 94
  and response time, 93, 94, 98
  and target location, 93, 94, 98

Sabotage graph, 17
SAFE, 13, 14, 15
  methodology, 13
Sample mean, 28, 44, 47, 48, 50
Sample size, 24, 31-32, 50, 73
  choice of, 31, 32
  and interval, 83
SEAD, 17
Seed, 36, 63, 90, 92
  and interval, 83
  and LB > MIN, 92
  and paths generated, 83, 92
SETS, 17
Set-up time, 98
Simulation, 14
  multiple target, 79, 92
  time for, 93
Standard deviation, sample, 28, 44, 47, 50
  of dump, 39
  of sample mean, 28, 50
Statistical tolerance interval, 30
Statistics, 22, 44, 46, 47, 48
  as estimates of population parameters, 50
  examples of, 28-31
Stochastic shortest route, 13, 101
Storage requirements, 99
Subroutine, 33
  DFLT, 36-37, 64
  EZ, 33, 37, 46-48
  MNDPT1, 33, 37, 41-44
  MOD, 37, 38
  PATHS, 33, 37, 38-39, 44-46
  PDIJK, 33, 39-41
  THYME, 33, 37, 46-48
  XPATH, 33, 36, 46, 47, 48

Target(s), 14, 17, 33, 42
  label, 33, 62
  location, 93, 94, 98
  number of, 33, 93
  response time, 33, 93, 98
Threshold value (P_T), 23, 24, 30-31, 87
THYME, 33, 37, 46-48
Time limit, 67
Time parameters, 19, 25
  initial values as modes, 19, 37
  randomness in, 15
  as random variables, 15, 22
Traversal time, 14, 23, 49
  conditional, 26, 27
  cumulative, 42, 47
  examination of, 64
  expected value and variance of, 46
  as measure, 20, 41, 46
  modification of, 64
  unconditional, 27, 47, 74
Type code, 36, 38, 54
  terminate modifications by, 38, 65, 66
  use of, 64-66

Unconditional probability, 25-26
  estimate of mean, 48, 49, 73, 74
  evaluation of, 26, 47, 48
  expected value for, 26, 47
Upper bound, 14, 47, 48, 73
  as conservative value, 87
  expected value for, 26
  MIN, 25, 28, 31, 47, 48
User inputs, 33, 36-38
  examples of, 53-67

Variance, 23, 25, 46
  analysis of, 90
  computation of, 50
  magnitude of, 25n

Weight, arc, 17, 19
  as detection probability, 17, 19, 40
  node, 17
  as traversal (delay) time, 17, 19

XPATH, 33, 36, 46, 47, 48

GLOSSARY

FACILITY GRAPH. A representation of a facility using nodes and arcs.

FACILITY THRESHOLD. An arbitrary value which is assigned to the measure (see below) in order to designate an acceptable security threshold for the facility.

GENERATED PATH. For a given realization (see below), a path that is a minimum with respect to the measure.

GUARD RESPONSE TIME or RESPONSE TIME. The time required for a guard to move from his present location (or station) to the target site.

IRUN. The number of user-specified evaluating (Phase 2) replications for each generated path.

LOWER BOUND (LB). In any simulation, a minimum interruption path from among all possible paths to all specified targets is generated in each pathfinding replication. The lower bound is the mean of the interruption probabilities for these generated paths.

MEASURE. The criterion by which a path is determined to be a minimum.

MRUN. The number of user-specified pathfinding (Phase 1) replications.

REALIZATION. A set of values for the input parameters obtained by Monte Carlo sampling of the prior probability distributions for these parameters.

REPLICATION or RUN. One pathfinding search (Phase 1 run) or one evaluation of a (generated) path (Phase 2 run). Each search or evaluation is associated with a new realization or set of values for the input parameters.

RUN TIME. The number of central processor (CP) seconds required to carry out a simulation (not to be confused with a time per replication).

SAMPLE SIZE. The number of replications in the pathfinding phase (Phase 1) or the evaluation phase (Phase 2) of a simulation.

SIMULATION. One complete PATHS execution consisting of a specified number (MRUN) of pathfinding replications and a specified number (IRUN) of evaluations for each generated path.

TARGET. The object of sabotage or theft, which is represented in the facility graph by a specified terminal node (path endpoint).

TRAVERSAL TIME. Depending upon the context, the time required to traverse a node, an arc, or a path. Node and arc traversal times are sometimes called "task" or "delay" times when referring to an input. A path traversal time can only refer to an output of a simulation.

TYPE I and TYPE II ERRORS. If a statistical hypothesis is rejected when it should be accepted, a Type I error is made; if the hypothesis is accepted when it should be rejected, a Type II error is made.

UPPER BOUND or MINIMUM (MIN). For any simulation, the estimate of the mean interruption probability for the path that, among all generated (minimum) paths, has the least mean probability of interruption.

DISTRIBUTION:

U.S. NRC Distribution Contractor (CDSI)
7300 Pearl Street
Bethesda, MD 20014
320 copies for RS
25 copies for NTIS

Author selected distribution - 38 copies
(List available from author.)

400	C. Winter
1000	G. A. Fowler
1230	W. L. Stevens, Attn: R. E. Smith, 1233
1233	M. D. Olman
1700	W. C. Myre
1710	V. E. Blake, Attn: M. R. Madsen, 1714
1716	R. L. Wilde
1720	C. H. Mauney, Attn: J. W. Kane, 1721
1730	J. D. Kennedy, Attn: W. N. Caudle, 1734
1750	T. A. Sellers, Attn: M. J. Eaton, 1759
1751	J. J. Baremore, Attn: A. E. Winblad, 1751
1752	V. K. Smith
1754	I. G. Waddoups,
1760	J. Jacobs, Attn: M. N. Cravens, 1761, J. M. deMontmollin, 1760A W. F. Hartman, 1760A J. D. Williams, 1769
1762	H. E. Hansen
1768	C. E. Olson, Attn: G. A. Kinemond, 1768
1765	D. S. Miyoshi
2644	C. Pavlakos
4400	A. W. Snyder
4410	D. J. McCloskey
4413	N. R. Ortiz
4414	G. B. Varnado
4416	L. D. Chapman (15)
4416	K. G. Adams
4416	J. A. Allensworth
4416	H. A. Bennett
4416	L. M. Grady
4416	C. P. Harlan
4416	R. D. Jones
4416	M. T. Olascoaga
4416	J. M. Richardson
4416	S. L. K. Rountree
4416	D. W. Sasser
4756	D. Engi (8)
5000	J. K. Galt
5600	D. B. Shuster, Attn: A. A. Lieber, M. M. Newsom, 5620 R. C. Maydew 5630
5640	G. J. Simmons, Attn: R. J. Thompson, 5641 L. F. Shampine, 5642
5642	B. L. Hulme
8214	M. A. Pound
3141	L. J. Erickson (5)
3151	W. L. Garner (3) For: DOE/TIC (Unlimited Release)

☆U.S. GOVERNMENT PRINTING OFFICE:1981-0-777-023/627

Org.	Bldg	Name	Rec'd by *	Org.	Bldg	Name	Rec'd by